Norwegian University
of Life Sciences

Master's Thesis 2022 60 ECTS
Facutly of Biosciences

Exploring the mineral profile and transcriptomic responses to light stress in sugar kelp (*S. latissima*)

Simon Oscar Andersen Haughom
Biology

Acknowledgements

This thesis was written at the Faculty of Bioscioences, NMBU, Ås, and is part of the SafeKelp and Breed4Kelp2Feed projects. This research is funded by The Research Council of Norway.

I would like to thank my supervisors Åshild Gunilla Ergon, Mallikarjuna Rao Kovi, Franz Ronald Saavedra Goecke, and Liv Torunn Mydland. I have received valued guidance throughout the field and lab work, and during analysis and writing of the thesis. I would also like to thank Sylvia Sagen Johnsen for guiding me during the RNA-extraction at the CIGENE lab.

I would like to thank the employees at Seaweed Solutions AS for allowing us to collect samples from their harvest and for helping us out in the phenotyping work. A thank you also to members of the SafeKelp team at Sintef Ocean in Trondheim, Silje Forbord, Margot Nyeggen, and Åsmund Johansen, for the collaboration and allowing me to expand on the light stress experiment that they set up. I would also like to thank Mina Marthinsen Langfjord for the analyses performed at the Faculty of Environmental Sciences and Natural Resources Management (MINA, NMBU).

A workshop on bioinformatics arranged by Mallikarjuna Rao Kovi, Akhil Reddy Pashapu, Teshome Dagne Mulugeta, Marian Schubert, and Odd Arne Rognli helped me greatly in navigating the RNA-seq postprocessing.

Abstract

The macroalgae industry is growing and sugar kelp (*Saccharina latissima*) has become a popular choice for cultivation. One of the challenges for the species as a product for human consumption and animal feed is high contents of iodine and arsenic. Iodine is an important antioxidant in brown macroalgae, and decreased iodine content in *S. latissima* individuals could inflict reduced stress tolerance.

Mineral content was investigated in relation to thallus part, depth, thallus size, and ash percentage in cultivated *S. latissima*. A transcriptome analysis was performed on *S. latissima* samples from an earlier light stress experiment performed at Sintef Ocean Trondheim as part of the SafeKelp project.

Comparisons between stipe and bottom samples showed that many inorganic elements have different concentrations in the stipe and bottom (but not iodine). Iodine, arsenic, phosphorus, and vanadium levels were significantly correlated with rope section (depth) in bottom samples, and bromine and phosphorus were significantly related to ash content. Thallus size (fresh weight) tended to increase in individuals growing closer to the surface, but the relationship was not significant.

Proteins presumed to be vanadium-dependent iodoperoxidase (vIPO) and vanadium-dependent bromoperoxidases vBPO was upregulated in samples treated with 250 µmol $m^{-2}s^{-1}$ light, but efflux of iodine or bromine content was not observed in these samples. Arsenic and lead contents decreased with time at 250 µmol $m^{-2}s^{-1}$ light exposure. The metal complexing protein glutathione S-transferase was upregulated for samples exposed to 250 µmol light (days 1, 3, and 9). Various other stress related processes were observed to be regulated in the treated samples compared to control, including the xanthophyll cycle, base-excision repair, and peroxisome activity.

Table of Contents

Acknowledgements 1

Abstract 2

Introduction 4

Background 4

Literature review 4

Research questions and objective 11

Materials and methods 13

Study 1: mineral composition in cultivated *S. latissima* individuals 13

Study 2: Transcriptome analysis of light-stressed *S. latissima* individuals 17

Mineral composition 17

Transcriptome analysis 20

Results 28

Study 1: mineral composition in *S. latissima* individuals 28

Study 2: Transcriptome analysis of light-stressed *S. latissima* individuals 33

Analysis of mineral profile 33

RNA-seq results 38

Discussion 49

Study 1: mineral composition in *S. latissima* individuals 49

Subset 1 49

Subset 2 51

Study 2: Transcriptome analysis of light-stressed *S. latissima* individuals 52

Remarks on methods 52

Differentially expressed genes 54

Gene ontology analysis, gene products and mineral contents 55

Conclusion 61

References 62

Appendix I 69

Appendix II 78

Introduction

Background

Demand and production of cultivated brown macroalgae is increasing in Norway (Stévant et al., 2017), and sugar kelp (*Saccharina latissima*) is a popular cultivation choice. The species has many desired traits such as high content of nutrients and bioactive compounds (Nielsen et al., 2020) but there are concerns linked to high contents of iodine and arsenic. Because iodine in kelp is related to stress tolerance, there is a need to know whether efforts to reduce iodine content in living plants would inflict decreased stress tolerance. Uncertainty remains regarding iodine and arsenic levels in brown macroalgae used for food and feed, and further research is needed to guide EU law-making decisions (Petursdottir et al., 2015). As the macroalgae cultivation industry grows, there is an increasing need to study both beneficial as well as potentially harmful compounds and elements in cultivation species such as *S. latissima*.

Literature review

The macroalgae industry is growing

Macroalgae has historically been a valued food source especially in East Asian countries. The traditional uses for algae in Norway have been as feed and fertilizer (Goecke et al., 2020). Macroalgae today is used for food, feed, dietary products, pharmaceuticals, alginate production, biofuel, and more. To meet demand, there has been a substantial increase in total macroalgae production over the last 20 years. In 2016, 31.2 million tons fresh weight was produced globally, and production is expected to continue to increase (Goecke et al., 2020).

Macroalgae are habitat-forming species providing food and shelter to a variety of organisms. In addition, they help in energy transfer and nutrient cycling (Teagle et al., 2017). Chile, China, and Norway are leading the exploitation of wild macroalgae, and wild harvest is still the dominant practice in many countries (Buschmann et al., 2017). To ensure the health of ecosystems dependent on macroalgae, it is vital that future macroalgae exploitation depends on cultivation.

In 2015, Norway produced approximately 150,000 tons fresh weight macroalgae (mostly from wild harvest), making up 65% of the total European supply (EU 2018). The brown algae *Saccharina latissima* is a popular species for cultivation and has been successfully cultivated across Europe (Nielsen et al., 2020). *S. latissima* offers several beneficial nutritional compounds for humans such as minerals, phenolic compounds, sugars, and polyunsaturated fatty acids (Nielsen et al., 2020).

S. latissima is a powerful accumulator of iodine

Iodine in trace amounts is essential to human health, and insufficient iodine intake affects around 2 billion people globally (Krela-Kaźmierczak et al., 2021). On the other hand, too much iodine can be toxic and cause either hypothyroidism or hyperthyroidism (Leung & Braverman, 2014). The recommended daily intake of iodine is 150 µg day^{-1} for adults and adolescents according to FAO & WHO (FAO & WHO, 2001), and only 32 mg of *S. latissima* dry biomass is enough to meet daily recommended intake for adults (Roleda et al., 2018).

As far as we know, brown macroalgae are the most powerful accumulators of iodine of all living organisms. While seawater contains about 50 µg L^{-1}, *S. latissima* typically contains around 2.7-3.5 g kg^{-1} iodine dry biomass (Sharma et al., 2018), a difference in concentration amounting to 5 orders of magnitude. Iodine is found as iodide (I^-) in seawater and uptake is thought to be dependent on iodide oxidation in the apoplast producing I_2 or HOI (Küpper & Carrano, 2019). Specific enzymes belonging to a group called vanadium-dependent haloperoxidases (vHPOs) are likely to be responsible for this oxidation, catalysed by hydrogen peroxide (H2O2). There are three types of vHPOs called iodoperoxidases (vIPOs), bromoperoxidases (vBPOs) and chloroperoxidases (vCPOs). vCPOs appear in cyanobacteria but does not seem to be present in brown or red algae (Liang et al., 2014). The classification of these enzymes is based upon the most electronegative halogen that the enzyme can oxidise (not counting astatine), meaning that vCPOs can oxidise chlorine, bromine, and iodine, while vBPOs can oxidise bromine and iodine. vIPOs can only reduce iodine but is approximately 7 times more effective in this regard than vBPOs in *Laminaria digitata* (Colin et al., 2003).

Gametophytes in Laminariales, which lack vHPOs, does not seem to accumulate iodine (Küpper et al., 1998), which is strong evidence that these enzymes are central to the uptake of iodide from seawater. However, when exposed to H_2O_2, accumulation of iodine was initiated. Another pathway for iodine accumulation in Laminariales has been proposed to account for this, where I^- reacts with Fe(III) to produce I_2 (Crans et al., 2004), as seen in Equation 1.

$$Fe(III) + 2I^- \rightarrow 2Fe(II) + I_2 \qquad \text{Equation 1}$$

After oxidation of iodide to either I_2 or HOI (Equation 2 and 3), the increased lipophilic nature of these species allows them to pass through the cell membrane (Küpper & Carrano, 2019). After this point however, the details of I^- uptake are highly uncertain. HOI and I_2 are thought to be reduced back to I^- and stored in either the apoplast or vacuoles in the cell.

Release of iodine is an important stress response in S. latissima

S. latissima is found mainly in the subtidal zone and sometimes in the intertidal zone (Bekkby & Moy, 2011), an environment where these macroalgae is exposed to several stressors that define their population. Stressors include exposure to air and intense sunlight, high temperature, ice formation, sedimentation, exposure to strong waves, as well as a wide variety of biotic stressors such as herbivores, epiphytes, and pathogens. According to Maharana et al. (2015), formation of reactive oxygen species (ROS) is related to both temperature and sunlight. ROS formation events happen when light energy exceeds photosynthetic capacity in photosystem II. The process results among other things in the formation of singlet oxygen ($^1O_2$) (Pospíšil, 2016), a highly reactive species. Other ROS include superoxide anion radicals (O_2^-), hydroxyl radicals (OH^-), and hydrogen peroxide (H_2O_2) (Maharana et al., 2015).

One proposed stress response in Phaeophyceae is the scavenging of ROS by iodine, catalyzed by vIPO (Almeida et al., 2001). Once H_2O_2 or other ROS are present, iodide in the cells is oxidized and hypoiodous acid produced, while hydrogen peroxide is reduced to water (Equation 2 and 3). Hypoiodous acid reacts with a wide variety of organic substrates and forms iodocarbons. Iodocarbons can disturb bacteria biofilm formation and thus making them unable to form colonies. Bromide have also been found to act as an antioxidant, but iodide is thought to be more effective in this regard (Küpper et al., 2013).

$$I^- + H_2O_2 \rightleftarrows HIO + OH^- \qquad \textit{Equation 2}$$

$$HIO + I^- + H^+ \rightleftarrows I_2 + H_2O \qquad \textit{Equation 3}$$

Variations in iodine concentration

Iodine in *S. latissima* has been found to be relatively stable across years and seasons (Roleda et al., 2018). The same study looked at variation in iodine content in *S. latissima* between three geographical locations (France, Trondheim, and Bodø) and found significant differences. Regarding the distribution of iodine in *S. latissima* individuals we first need an introduction to *S. latissima* anatomy. A typical *S. latissima* individual has a holdfast connected to its substrate, and a stipe connecting the holdfast to a single, flat blade. The meristem is where non-differentiated cells are located which gives rise to other blade tissue by cell division and is located at the bottom close to the stipe. Early research on this topic suggested that iodine is located mostly in the holdfast, stipe, and meristematic zone, while the old part of the blade towards the tip has lower concentrations (Amat & Srivastava, 1985). This is in line with the finding that younger plants contain more iodine (Roleda et al., 2018). In 2008, Verhaeghe et al. showed that most iodine is stored in peripheral tissue, presumably in apoplastic spaces. Most of this iodine is stored as labile inorganic I^- (up to 90%), and the rest is stored in various iodinated amino acid residues such as iodotyrosine.

Bromine and vanadium-dependent bromoperoxidases (vBPOs)

Bromine has a very similar role to iodine in brown macroalgae, acting as a simple antioxidant. Despite this, not nearly as much is known about bromine in kelp as is known about iodine. Verhaeghe et al. (2008) found bromine in *Laminaria digitata* to be distributed in the apoplast of peripheral tissue, just like iodine. In another study by Küpper et al. (2013), the highest levels of bromine found in *L. digitata* individuals was found in the peripheral layers of the stipe and in the holdfast. Most bromine is present as bromide (Br^-) (Küpper et al., 2013). Even though iodine is the most efficient antioxidant in the presence of hydrogen peroxide, hydroxyl radicals and ozone (Colin et al., 2003; Küpper et al., 2008), bromine was found to the most effective antioxidant in the presence of superoxide (Küpper et al., 2013). Like vIPOs, vBPOs scavenge ROS producing halogenated compounds.

In addition, bromine and vBPOs are thought to be involved processes other than scavenging of ROS; in the interaction and cross-linking between macromolecules in another brown algae, *Fucus serratus* (Berglin et al., 2004), and in the disruption of quorum sensing (making bacteria unable to form biofilms) in red algae (Sandy et al., 2011).

Heavy metals

Arsenic, cadmium, and lead are heavy metals and are toxic even in trace amounts. Cell wall chemistry is the main factor that decides uptake rates of heavy metals in algae. While cadmium is mainly sequestered into algae by chelation, arsenic is taken up by adsorption and active uptake, and lead uptake requires chelation as well as ion exchange and reduction reactions (Roleda et al., 2019). The main cell wall polysaccharides associated with heavy metal uptake are carboxyl, amino, sulfhydryl, and sulfonate (Roleda et al., 2019).

Previous research on heavy metals in *S. latissima* has found variation across seasons to be small, except for Hg having higher levels in autumn (Roleda et al., 2019). The same study reported variation across geographical locations to be significant for As, Cd, Hg, and Pb. This suggests that heavy metal content in brown macroalgae is highly dependent on the local environment.

There is more concern around arsenic levels in *S. latissima* than around cadmium and lead levels. Arsenic appears as both inorganic arsenic (i-As) and as part of a variety of organic compounds such as arsenosugars, arsenolipids, and arsenosugarphospholipids. There is great complexity among the organic compounds that is not yet accounted for. Regarding arsenic distribution in individuals of *S. latissima*, the stipe appears to have lower content of total arsenic (Pétursdóttir et al., 2019). The same study also found that arsenic speciation changed with tissue age in *S. latissima*, specifically that older tissue parts (such as the tip of the blade) contained more water-soluble arsenic than younger tissue (meristematic region).

Despite the complexity among the organic arsenic compounds, i-As seems to be the speciation of arsenic most associated with toxicity (Petursdottir et al., 2015). According to Roleda et al. (2019), only 1,6% on average of the total arsenic found in brown macroalgae is i-As. The same study conducted a health risk assessment and found that human consumption of *S. latissima* can generally be considered safe for seaweed harvested from non-contaminated waters. In France, the maximum allowable level of inorganic arsenic in algae condiments is 3 mg/kg (Petursdottir et al., 2015), but there is still no EU regulation concerning maximum arsenic levels in seaweeds (European Commission, 2015), partly because of the complexity of speciation of arsenic in seaweeds.

Glutathione is involved in heavy metal protection in brown algae

Glutathione is a highly conserved peptide found in almost all parts of the plant cell, yet the highest concentrations are found in the chloroplasts (Ahmad et al., 2010), where it plays an important role in the glutathione-ascorbate antioxidant system (Caverzan et al., 2012). Glutathione is found in its reduced form most of the time and can be linked to xenobiotics such as heavy metals by the glutathione S-transferase enzyme. Glutathione reductase is responsible for reducing oxidised glutathione back to its reduced form (Nowicka, 2022).

Earlier research shows that glutathione combats heavy metal toxicity in the brown algae *Fucus serratus* and *Fucus vesiculosus,* by complexing metals including arsenic, cadmium and lead (Pawlik-Skowrońska et al., 2007). In addition, glutathione is the primary precursor to phytochelatins, another protein capable of complexing metals, and glutathione content was found in the same study to be a limiting factor in phytochelatin production.

Methods for reducing potentially harmful elements after harvest

Water blanching has been proposed as a method to reduce iodine content in *S. latissima* (Nielsen et al., 2020). Biomass soaking is another proposed method, which has been shown to reduce iodine content in *S. latissima*, and to reduce cadmium content in *Alaria esculenta*. However, these methods both results in collateral loss of beneficial compounds, as well as changing texture, colour, and taste.

The possibility of using cultivar development to address high iodine and arsenic levels has not yet been explored. Cultivar development of kelp has focused mostly on biomass and size until now (Goecke et al., 2020), and attempts at targeting chemical composition has been focused on increasing beneficial compounds (Zhang et al., 2007).

RNA-sequencing

The term RNA-sequencing or RNA-seq was first coined in 2008 (Nagalakshmi et al.) after being developed for 2-3 years and has since revolutionized transcriptomics. RNA-seq is characterized by the conversion of isolated RNA into double-stranded complementary DNA (ds-cDNA) by reverse transcription, which is then sequenced by various PCR methods (Weber, 2015). The technology offers several benefits over methods like Sanger sequencing and microarray-based sequencing. Firstly, RNA-seq is not based on already known sequences like in microarray methods and can thus be used to discover novel sequences. RNA-seq also provides higher coverage and greater resolution than both Sanger and microarray-based approaches (Kukurba & Montgomery, 2015). The RNA-seq technology is still developing today with improvements arriving continuously.

There is a variety of tools available for the purpose of connecting sequenced reads to previously discovered genes and gene products in annotated genomes. The computing process of aligning transcript reads to already known protein sequences is known as blastx. The recently developed Diamond software claims faster execution than previous blastx softwares yet similar sensitivity (Buchfink et al., 2021). The Gene Ontology Consortium (GOC) started out as an endeavour to gather genes related to core biological functions in a common database (Ashburner et al., 2000), and has since become a standard resource for transcriptome studies when it comes to gene function annotation (Carbon et al., 2021).

Two brown algae species have complete genome sequences at this point: *Ectocarpus siliculosus* (Cock et al., 2010) and *Saccharina japonica* (Ye et al., 2015). Ref focused thoroughly on vIPOs and vBPOs in their sequencing of the *S. japonica* genome. Until the complete *S. latissima* genome is sequenced, transcriptome projects on *S. latissima* must depend on these genomes.

Previous transcriptome studies on brown algae

The physiology and ecological requirements of *S. latissima* has been studied for several decades, including research on the effects of salinity and irradiance on growth (Spurkland & Iken, 2011) and the effects of visible and ultraviolet light on growth and damage (Davison et al., 2007; Roleda et al., 2006, 2007; Wiencke et al., 2006). Despite this, the molecular biology and genetics of *Laminariales* is still not well understood (Heinrich, Frickenhaus, et al., 2012; Küpper & Carrano, 2019).

During the early years of RNA-seq, Colin et al. (2003, 2005) looked at vHPO genes in *L. digitata* and found that vBPOs and vIPOs share a close common ancestor and that both vBPOs and vIPOs are encoded by a multigenic family. In 2014, (Liang et al., 2014) did a transcriptome analysis on *S. japonica* to show that vIPOs and vBPOs are closely related and likely diverged from vCPOs during an earlier time, presumably during the period of prokaryotic life.

(Heinrich, Frickenhaus, et al., 2012) performed RNA-seq on *S. latissima* and in the process described a protocol for *S. latissima* RNA extraction which was used as a basis for the RNA extraction performed in this study. In the transcriptome study (Heinrich, Valentin, et al., 2012), they used a microarray analysis to investigate acclimation processes in light and temperature stressed *S. latissima*. They found high temperature to have a bigger impact on the transcriptome than high light. Monteiro et al. (2019) used RNA-seq to study the role of geographical variation in the stress response of *S. latissima* to temperature and salinity. They found that, depending on the geographical origin of the specimens, different transcriptomic responses were observed after combinations of temperature and salinity exposure.

Research questions and objective

The objective of this research is to evaluate whether future attempts to reduce iodine and arsenic content in *S. latissima* is likely to compromise stress tolerance. This is not research on breeding, but an attempt to guide future breeding research on how stress tolerance possibly relates to the mineral composition in *S. latissima*. Several research questions were constructed as part of this objective:

i) Variations in inorganic elements in *S. latissima*
   a. Are there differences in content of inorganic elements between the bottom of the blade and the stipe?
   b. Is there a relationship between rope section and content of inorganic elements?
   c. Is there a relationship between thallus size and content of inorganic elements?
   d. Is there a correlation between ash content and content of inorganic elements?

ii) Changes in gene expression as result of light-stress in *S. latissima*

   a. Which genes and processes are regulated by light-stress?
   b. Are there changes to iodine content and genes associated with iodine metabolism in response to light stress?
   c. Are there changes to arsenic, cadmium, and lead and genes associated with heavy metal in response to light stress?

I aim to compare different parts of the thallus to know whether a sample from one part of the thallus can be used to predict mineral contents in other parts of the thallus. The original plan was to use samples from the middle and top of the blade as well, but because of challenges with the sample material, only the stipe and bottom part of the blade were examined. As it is desirable to select for large blade size in breeding, I also aim to check for correlation between thallus size and chemical profile. There is also the aspect that any change in the content of a single element could simply be the result of a change in all elements (that is, an increase or decrease in ash content). This needs to be kept in mind when looking at potential correlations between mineral contents and rope section and thallus size. I aim to investigate which transcriptomic processes are regulated during light stress, and especially processes related to iodine and heavy metals.

Materials and methods

Study 1: mineral composition in cultivated *S. latissima* individuals

Sampling

The algae are grown on 15-meter-long ropes stretched out between buoys in the ocean. Individuals used in this experiment were cultivated on ropes at the cultivation site of Seaweed Solutions off the island of Frøya at the end of the Trondheim fjord. They were harvested in June 2021 and transported to the distribution facility of Seaweed Solutions in Trondheim where they were phenotyped.

Design and phenotyping

The samples arrived in boxes of about 2 m^2, each box containing one rope of 15 m. For each rope, the first 0.5 m was cut off. Then sections of 1.5 m were cut off from the rope. The longest individual was phenotyped first, then starting from one end of the rope section 19 other individuals were phenotyped. Fresh weight was measured for each individual.

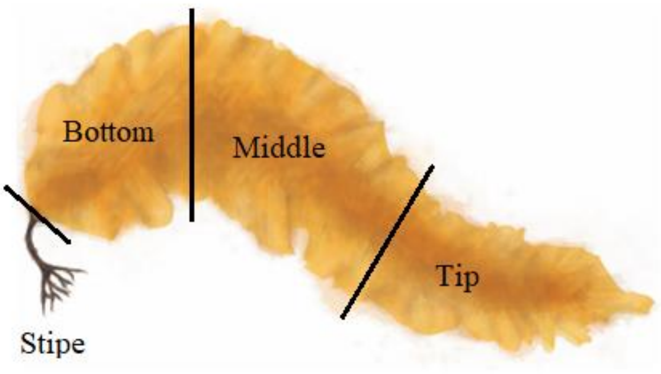


Figure 1: Separation of thallus part samples used for mineral profiling in S. latissima*. Picture painted by* Lydia Gieselmann (n.d.) *on commission for Silje Forbord.*

In addition, individuals of more than 5 g fresh weight from sections 5-9 on rope nr. S2 were selected for investigating relationships between mineral profile, thallus parts, thallus size, rope section and ash content. Thalli were separated into stipe, bottom, middle and top part samples as shown approximately in Figure 1, then each sample was weighed. These samples were delivered to Sintef Ocean Trondheim for freeze-drying.

Bryozoa score

A discrete "Bryozoa score" was noted after the freeze-drying step for each of the samples to be analysed. This was based on a rough eye-test, and the samples were scored from 1-4 as shown in table ?.

Table 1: Scoring system for bryozoa contamination on freeze-dried S. latissima *samples. Only the samples scheduled for mineral analysis were scored. The scoring scale ranges from 1-4 where 1 is low amount bryozoan fouling and 4 is high amount of bryozoan fouling.*

Score	*1*	*2*	*3*	*4*
Description	0-2 colonies	3-4 colonies	5-6 colonies	7+ colonies

Analysis of mineral content

Then, the freeze-dried samples were ground in liquid nitrogen (LIN) and funnelled into 15 ml tubes. LIN was not strictly necessary as the samples were already freeze-dried but it made the samples tissue brittle and easier to crush. For the largest samples, 0.1 – 0.2 g were set aside for ash percentage analysis. Then the samples were sent to an analysis laboratory belonging to the Faculty of Environmental Sciences and Natural Resources Management (MINA) at NMBU and analysed by Mina Marthinsen Langfjord for contents of iodine, arsenic, bromine, phosphor, vanadium, cobalt, cadmium, and lead. Iodine and bromine were analysed after basic extraction by Inductively Coupled Plasma Mass Spectrometry (ICP-MS), and the other elements were analysed after HNO_3 decomposition in an UltraClave (MLS Milestone, Italy) at 260 °C using ICP-MS.

Analysis of ash percentage

Ash percentage was measured by incinerating the samples in heat-resistant tubes overnight at 550 °C. Samples were weighed before and after incineration, and the ash percentage calculated by dividing the weight of the ash by the weight of the original sample. Many samples (especially stipe samples) did not have enough dry weight to cover both analysis of elements and ash percentage analysis. Therefore, ash percentage was measured mostly in bottom blade samples as they were generally larger.

Section as a measure of depth

As shown in Figure 2, the first section used was approximately in the middle of the rope (sections 5), and sections 6, 7 and 8 continued towards one end of the rope. The middle of the rope would naturally be hanging lower, although it is uncertain how much lower exactly.

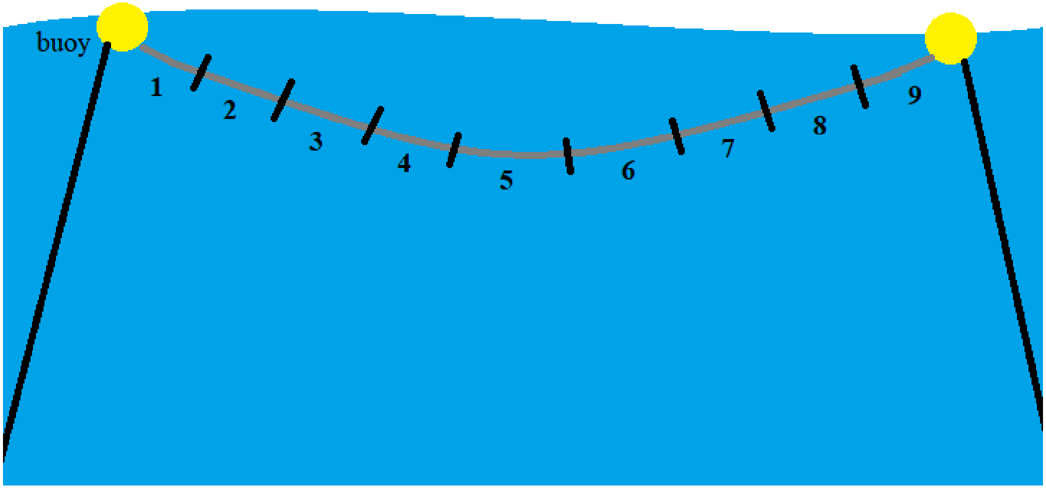


Figure 2: Approximate depth placement of rope sections on which S. latissima *samples were grown. The gray line is the rope stretched out between two buoys (yellow). This figure only illustrates approximate depth placements of rope sections 5-8 used in this experiment.*

Statistical analysis

Statistical analyses were performed using R Studio (R Team, 2021) and Microsoft Excel.

Three different subsets of the original dataset were used. The first subset (subset 1) contains 20 samples from 10 individuals where every individual has one stipe sample and one sample from the bottom of the blade. The second subset (Subset 2) contains only samples from the bottom and includes some individuals that did not have a matching stipe sample and so could not be included in the first subset. The third subset (Subset 3) contains all possible samples that are from different individuals (i.e., thallus part did not matter, only rope section and measured wet weight of the original individual.

A Principal component analysis (PCA) was carried out for each Subset. Variables included in the PCA for Subset 1 was the 8 element variables. In the PCA for Subset 2, ash percentage was included in addition to the 8 element variables.

A 2-way analysis of variance (ANOVA) was performed for 8 elements: iodine, arsenic, bromine, phosphorus, vanadium, cobalt, cadmium, and lead. I looked at the main effects of depth and part but not interaction as shown in (Equation 4).

$$Element_{ijk} = \mu + Depth_i + Part_j + \varepsilon_{ijk}$$

Equation 4

Before arriving at the model described above, some other models were tested. A 3-way ANOVA model was tried for Subset 1 including individual as a factor nested in rope section, as well as the main effect of thallus part. Individual did not show any effect in this model and was therefore excluded going forward. A complete 2-way model was also attempted, including the effect of interaction in addition to the main effects of section and part. As only vanadium was significantly affected by the interaction, the complete model was also dropped in favour of the main effects model.

Study 2: Transcriptome analysis of light-stressed *S. latissima* individuals

Mineral composition

Sampling

Individuals were cultivated on ropes at a Seaweed Solutions farm off the island of Frøya and sampled in May 2021 by Silje Forbord and Margot Nyeggen. *S. latissima* individuals with length of about 60 cm were chosen. After harvesting, the thalli were transported to Sintef Ocean Trondheim and an acclimation period was begun as explained in the section below.

Experimental design

The experiment was a crossed design experiment in which tissues were treated with either 100 or 250 µmol $m^{-2}s^{-1}$ light for 1, 3 or 9 days. Two tissue pieces (15 cm each) from each sporophyte were cut from the central area above the meristem (Figure 3) and distributed indiscriminately in beakers. They were acclimated in flow-through deep water at 10 °C exposed to 40 µmol light. In addition, 4 whole sporophytes were acclimated the same way. The acclimation lasted for 7 days until the experiment began.

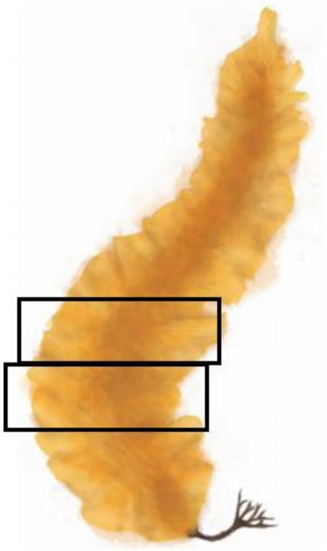


Figure 3: Separation of thallus part samples used for mineral profiling in S. latissima. *Picture painted by* Lydia Gieselmann (n.d.) *on commission for Silje Forbord.*

As can be seen in Table ?, there were 6 treatment groups (100 day 1, 250 day 1, 100 day 3, 250 day3, 100 day 9, and 250 day 9) and 5 control groups exposed to the same light as in the acclimation period (day 0 uncut, day 0, day 1, day 3, day 9). All treatments and controls had 4 replicates, resulting in a total of 44 samples.

Table 2: Experimental setup for transcriptome study on light-stressed S. latissima *individuals. Abbreviations: C0=control day 0, C1=control day 1, C3=control day 3, C9=control day 9, MED1=samples exposed to 100 µmol m^{-2}s^{-1} light for 1 day, MED3=samples exposed to 100 µmol m^{-2}s^{-1} light for 3 days, MED9=samples exposed to 100 µmol m^{-2}s^{-1} for 9 days, MAX1=samples exposed to 250 µmol m^{-2}s^{-1} light for 1 day, MAX3=samples exposed to 250 µmol m^{-2}s^{-1} light for 3 days, MAX9=samples exposed to 250 µmol m^{-2}s^{-1} light for 9 days.*

	Day 0	*Day 1*	*Day 3*	*Day 9*
MIN (40 µmol light)	C0 and UC (uncut)	C1	C3	C9
MED (100 µmol light)		MED1	MED3	MED9
MAX (250 µmol light)		MAX1	MAX3	MAX9

At the end of the acclimation period fresh weight was measured for all samples. Then for each control sample, one piece was freeze-dried and saved for mineral content analysis and another part was frozen in liquid nitrogen and stored in a – 80 °C freezer, to be used in the RNA-seq analysis. At days 1, 3 and 9 of treatment the same was done for each sample due for that day. In addition, fresh weight was measured for the day 9 samples before freezing, to check whether different light intensities had effect on growth. Water samples for mineral analysis were collected at the start of the acclimation period, at the end of the acclimation period (day 0) and at the end of the experiment ? (day 9). Silje Forbord and Margot Nyeggen did the experimental setup of the beaker glasses, and Åsmund Johansen setup the light systems.

Analysis of minerals

The freeze-dried samples were handled together with the freeze-dried samples from Study 1 (see above), and analysed for the same mineral elements (iodine, arsenic, bromine, phosphor, vanadium, cobalt, cadmium, and lead) at an analysis laboratory belonging to the Faculty of Environmental Sciences and Natural Resources Management (MINA) at NMBU (analysed by Mina Marthinsen Langfjord).

Statistical analysis of minerals content

Mineral contents were analysed using R Studio and Microsoft Excel. Firstly, a Principal component analysis (PCA) was performed for the 8 element variables to get an overview of the variation. Day 0 samples were not included.

Several analyses of variation (ANOVAs) were carried out. Firstly, a one-way ANOVA was used to determine the effect of cutting the material as shown in Equation 5.

$$Element_{ij} = \mu + Cut_i + \varepsilon_{ij}$$

Equation 5

Another control was performed to determine the effect of time using 4 timepoints: one from the end of the acclimation period (excluding uncut samples), as well as day 1, 3 and 9 from the 40 µmol treatment.

$$Element_{ij} = \mu + Time_i + \varepsilon_{ij}$$

Equation 6

Lastly, a two-way complete ANOVA model was performed looking at the effects of both time and light (excluding all day 0 samples).

$$Element_{ijkl} = \mu + Time\ _i + Light_j + (Time * Light)_{ij} + \varepsilon_{ijk} \qquad \textit{Equation 7}$$

Transcriptome analysis

RNA extraction

The study done by (Heinrich, Frickenhaus, et al., 2012) was used a starting point for the RNA-extraction protocol, which was followed except for some modifications.

Before starting the protocol, buffer RPE was diluted with 4 volumes of ethanol 100 % (44 ml). In addition, the DNase I stock solution was prepared for on-column DNase extraction to be done later. Lyophilized DNase I (1500 Kunitz units) was dissolved in 550 µl of RNAse-free water. The stock solution of DNase I was removed from the vial and aliquoted into 1.5 ml tubes and stored in a – 20 °C freezer.

Between 100-150 mg of frozen sample was transferred to 2 ml sample tubes containing QIAGEN tungsten carbide beads and ground using TissueLyser (4x 30 seconds at 25 Hz). Samples were kept frozen using liquid nitrogen until the extraction buffer was added. One ml extraction buffer (100 mM Tris pH 8, 1 M NaCl, 50 mM EDTA pH 8, 2 % CTAB) were added to the tubes as well as 20 µl 2 M DTT. Samples were vortexed thoroughly then incubated at 45 °C for 10 minutes. One ml of chloroform:isoamylalchohol (24:1) were added and the tubes were shaken at maximum speed for 10 minutes, then centrifuged for 20 minutes (12000 x g, 20 °C). 750 µl of the aqueous phase were transferred to new 2 ml tubes. 1/3 volumes (mostly 225 µl) of 100 % EtOH were added, and tubes were turned upside down carefully to mix the samples. 1 volume (mostly 975 µl) of chloroform:isoamylalchohol (24:1) were added and the samples were centrifuged for 20 min (12000 x g, 20 °C). 600 µl of the aqueous phase were transferred to new tubes and 1 volume (600 µl) of chloroform were added. Samples were centrifuged for 10 min (12000 x g, 20 °C). 450 µl of the aqueous phase were transferred to new tubes.

From there, the QIAGEN RNeasy kit was used to continue the extraction according to the protocol supplied by the manufacturer (pp. 62 – 66), starting at step 3 (QIAGEN, 2012). 450

µl buffer RLT was added to each sample tube, and then tubes were vortexed vigorously. The lysate in each sample tube was transferred to QIAshredder spin columns placed in 2 ml collection tubes, then centrifuged for 2 min at 12000 x g, 20 °C. As the lysate exceeded 650 µl which is the limit of the QIAshredder spin columns, this operation was performed in two steps. The supernatant of the flow-through for each sample was transferred to a new 2 ml collection tubes. There was no visible pellet in the bottom the tubes because most debris was already removed in the chloroform extraction, but the bottom the tubes were not disturbed nevertheless. 0.5 volumes of 100 % ethanol (normally 400 – 450 µl) were added to the lysate and mixed by pipetting. Tubes were not vortexed or centrifuged at this point. The lysate was transferred to RNeasy spin columns placed in 2 ml collection tubes. The samples were centrifuged for 15 s at 8000 x g (20 °C), and the flow-through discarded. This step was performed twice for each sample (but gathered in the same spin column) because sample lysate volumes exceeded 650 µl.

On-column DNase digestion was performed at this point, following the same protocol (pp. 82-84). Firstly, 350 µl buffer RW1 was added to the RNeasy spin columns. Samples were centrifuged for 15 s at 8000 x g (20 °C), and the flow-through was discarded. A DNase I incubation mix was made by mixing 10 µl DNase I stock solution with 70 µl buffer RDD for every sample. Then 80 µl incubation mix was added to each sample by pipetting it directly onto the spin column membrane, and samples were left on the benchtop (~22 °C) for 15 min. After incubation the spin column membrane was washed by adding 350 µl buffer RW1 to the spin columns and centrifuging for 15 s at 8000 x g (20 °C). The flow-through was discarded.

Returning to the main protocol step 7 was skipped as this is another RW1 wash step. Then, 500 µl buffer RPE was added to the RNeasy spin columns. The centrifugation speed and length in this step was modified during the course of the project but started out 8000 x g for 15 s (20 °C) as stated in the protocol. However, to address some impurities detected in the final product, the speed and length was increased to 8500 – 9000 x g for 18 – 20 s (20 °C), and the step was repeated for some runs. In the next step another 500 µl buffer RPE was added to the spin columns and samples centrifuged for 2 min at 8000 x g (20 °C) to further wash and dry the spin columns. The spin columns were placed into new 2 ml collection tubes and the old collection tubes containing flow-through were discarded. The samples were then centrifuged for 10 min at 12000 x g (20 °C). The spin columns were place in new 1.5 ml collection tubes (supplied in the kit), and 30 µl RNAse-free water was added directly onto the spin columns. The samples were left on the benchtop for 1 min before centrifugation. They were centrifuged for 1 min at 8000 x g (20 °C), resulting in the final eluate as flow-through. For each sample a 3.5 µl aliquot was set aside for NanoDrop and TapeStation testing. NanoDrop helps assess both quantity and purity of the RNA samples (Thermo Fisher Scientific Inc., 2008), while the TapeStation analyses quantity and integrity of the RNA (Agilent Technologies Inc., 2015). Samples were stored in a – 80 °C freezer until they were shipped to NovoGene for RNA-sequencing.

Quality control, De novo assembly and reads alignment

After the raw sequences returned from NovoGene, the data was processed using a variety of command-line software tools. The complete pipeline for RNA-seq postprocessing I used is outlined below.

Firstly, the reads were checked for quality. FastQC provided an HTML report for each sample containing graphs showing potential areas of low quality (Andrews, 2010). To make interpretation of multiple samples easier, a program called MultiQC were used to compile the FastQC reports into a single HTML report (Ewels et al., 2016). A tool called Trimmomatic was used to trim areas of low quality from the sample reads (Bolger et al., 2014). Another round of FastQC/MultiQC was performed after trimming to check the quality again. The final MultiQC plots and stats can be seen in the appendix.

An essential part of the analysis is construction of a reference genome. If the genome of the organism is already fully sequenced, this genome can simply be used as reference. However, if a reference genome is not available, a "de novo assembly" can be constructed using samples from the experiment. A software program called Trinity was used to create the assembly in this study (Grabherr et al., 2011; Haas et al., 2013). Trinity is split into three software modules: Inchworm, Chrysalis and Butterfly.

The first module (Inchworm) compresses the data by a series of steps. Firstly, reads are decomposed into k-mers, meaning all possible substrings (nucleotide sequences) of length k are found and cataloged. K-mer length of 25 was chosen for this analysis. The k-mer that appears the most times is used as a seed k-mer and extended at the 3' end. Extension is determined by coverage, meaning the nucleotide that appears the most times in the catalogue of k-mers is added. If there are two or more nucleotides tied for most abundant, each path is explored to find which path offers the most cumulative coverage. Afterwards, the reverse path is extended by the same method. When no k-mers can be extended in either direction, the assembled sequence (called contig) is stored and all k-mers found in the contig removed from the k-mer catalogue. Then a new seed k-mer is found and the process repeated.

Chrysalis, the second module, takes contigs reported by Inchworm and groups them based on k-1mer (in this case 24-mer) overlaps. A k-mer graph (de Bruijn graph) is created for each group which shows overlaps between kmers and branches where there is variation.

In the last step, Butterfly, the de Bruijn graphs generated in Chrysalis are simplified. Stretches that contain no branching are collapsed. Sequencing reads from the trimmomatic output is threaded into the graph and the most likely paths through the graph is reported back. The result is linear full-length transcripts for all alternatively spliced isoforms.

After the reference genome was assembled Kraken2 was used as a taxonomic classifier (Wood et al., 2019). The database used is named “Standard” and contains sequences for archaea, bacteria, viral, plasmid, human, UniVec_Core (kraken2 page). Kraken2 queries the sequences to the database and outputs both classified and unclassified sequence files. The classified files contain sequences that matched to the database and thus contains sequences from other organisms.

BUSCO v5 was used to check the completeness of the assembly genome (Manni et al., 2021). The database for stramenopiles was chosen which contains 100 BUSCO markers. The BUSCO program compares gene content in the assembly to the expected gene content based on the chosen taxonomical database (in this case the stramenopiles database).

For aligning the reads to the de novo assembly, bowtie2 was used with RSEM as estimation method (Langmead & Salzberg, 2012; B. Li & Dewey, 2011). These tools try to match read sequences onto a reference genome, in this case the assembly that was made using Trinity. The end-to-end option was used, meaning the software tries aligning all the read characters in contrast to “local mode” where non-aligning end of reads can be trimmed to increase alignment score.

After alignment about 550000 reads remained. Unfortunately, the alignment output log file was lost, so the exact number of aligned reads is not known. A software program called Corset was used to cluster these reads into 178827 “clusters” based on similarity which were used in downstream analyses.

Identifying differentially expressed genes

The statistical analysis was performed in R Studio and Microsoft Excel. Functions from several R packages were used. All functions used are listed in Table ? along with their respective R packages and package citations. Complete scripts can be found in the appendix.

Table 3: Overview of R packages used in identification of differentially expressed genes, with corresponding citations.

R package	Functions	Package citation
edgeR	cpm(), filterByExpr(), calcNormFactors(),estimateGLMCommonDisp(), estimateGLMTrendedDisp(), estimateGLMTagwiseDisp()	(Chen et al., 2016; McCarthy et al., 2012; Robinson et al., 2009)
limma	plotDensities(), makeContrasts()	(Ritchie et al., 2015)
DESeq2	vst()	(Love et al., 2014)
waldo	compare()	(Wickham, 2022)
ade4	dudi.pca()	(Bougeard & Dray, 2018; Chessel et al., 2004; Dray et al., 2007; Dray & Dufour, 2007; Shanmugam, 2020)
ggplot2	ggplot()	(Ginestet, 2011)
stats	model.matrix()	(R Team, 2021)
seqinr	import.fasta(), write.fasta()	(Charif et al., 2015; Pearson & Lipman, 1988)
tidyr	drop_na()	(Wickham & Henry, 2019)
dplyr	left_join(), group_by(), summarize(), ungroup(), distinct(), slice_max(),	(Wickham et al., 2019)
Stringr	str_detect()	(Wickham, 2019)

RNA-seq data: import and data wrangling

The clusters from the Corset program containing the reads were imported into R Studio and assigned metadata. Rows in the data and metadata were compared to make sure they were equal (compare function), and then imported into a list (DGEList function). Counts per million mapped reads (CPM) was calculated (cpm function). CPM is defined as:

$$CPM = \frac{Number\ of\ reads\ mapped\ to\ gene * 10^6}{Total\ number\ of\ mapped\ reads} \qquad \text{Equation 8}$$

The density of counts (logCPM) was then plotted for all samples in a single plot (plotDensities function). Based on the plot, many transcripts contained low read count. Transcripts with a read count of less than 20 were consequently filtered out (filterByExpr function). Then, count density was plotted again using the filtered transcripts. Density plots are included in the appendix.

Library sizes were plotted against samples (shown in appendix). As the plot showed that there were imbalances in the library sizes, normalization factors were calculated to scale library sizes across samples (calcNornFactors function). The TMM method was used (weighted trimmed mean of M-values), proposed by Robinson & Oshlack (2010).

Next, a variance stabilizing transformation (VST) was performed (vst function), which is a useful way of checking for outliers. The VST is calculated from the fitted dispersion-mean relations (Anders & Huber, 2010). By dispersion in edgeR is meant the squared Biological Coefficient of Variation (BCV^2). The BCV is a factor, and between RNA sample replicates, the true, unknown gene abundance varies by this factor (McCarthy et al., 2012). The variance for the VST transformed counts was calculated and subsequently the 500 most variable genes were sorted out to be used in a PCA plot. The PCA was performed (dudi.pca function) and plotted (ggplot function).

The design matrix was made using grouped variables (model.matrix function), that is using treatment groups as the single variable instead of using light intensity and treatment duration as two different variables. The 0 in the model removes intercept (Equation 9).

$$Design = 0 + group_i + \varepsilon_{ij} \qquad \text{Equation 9}$$

Common dispersion, trended dispersion and tagwise dispersion were calculated (estimateGLMCommonDisp, estimateGLMTrendedDisp, and estimateGLMTagwiseDisp functions), and plotted against the average log CPM. The BSV was calculated by taking the square root of the estimated common dispersion.

The contrasts were made using the makeContrasts function. To best investigate the effect of both light-stress and treatment duration, the contrasts shown in Table 4 were chosen. The other contrasts were not investigated in this study.

Table 4: Statistical setup for transcriptome study on light-stressed S. latissima *individuals. Abbreviations: C0=control day 0, C1=control day 1, C3=control day 3, C9=control day 9, MED1=samples exposed to 100 µmol m^{-2}s^{-1} light for 1 day, MED3=samples exposed to 100 µmol m^{-2}s^{-1} light for 3 days, MED9=samples exposed to 100 µmol m^{-2}s^{-1} light for 9 days, MAX1=samples exposed to 250 µmol m^{-2}s^{-1} light for 1 day, MAX3=samples exposed to 250 µmol m^{-2}s^{-1} light for 3 days, MAX9=samples exposed to 250 µmol m^{-2}s^{-1} light for 9 days. All controls were exposed to 40 µmol m^{-2}s^{-1} light.*

	End of acclimation	*1 day*	*3 days*	*9 days*	*Comparison*
MIN (40 µmol light)	C0 and UC	C1	C3	C9	C9vsC0
MED (100 µmol light)		MED1	MED3	MED9	MED9vsMED1
MAX (250 µmol light)		MAX1	MAX3	MAX9	MAX9vsMAX1
Comparison	C0-UC	MAX1vsC1	MAX3vsC3	MAX9vsC9	

For every gene, a negative binomial GLM (generalized linear model) was fitted to read counts (glmFit function). This fit was then used to conduct genewise tests for each contrast.

Functional Analysis

Output tables containing the differentially expressed genes for each contrast were made. However, they were identified by Corset IDs, while the blastx algorithm requires FASTA files (containing the actual sequences). Using R to subset data, the FASTA sequences were extracted from the Trinity assembly by Trinity IDs and their corresponding Corset IDs. The seqinr package was used to read and write FASTA files (import.fasta and write.fasta functions).

Then, blastx was run using the diamond software (Buchfink et al., 2021) on default sensitivity (finds hits of more than 60% identity). This program annotates proteins to the sequences by querying the sequences to a database. The “outfmt” output format value was set to 5, because Blast2Go (explained below) requires input files as .xml files.

Subsequently, Blast2Go was used to make annotation maps and to annotate GO terms to the blastx hits. (Götz et al., 2008). When importing the blast results into Blast2Go, Highest Scoring Pair (HSP) Length Cutoff was set to 33, and HSP-Hit Coverage was set to 80. The latest Goa database version at the time was used (2022.03) for annotation mapping. For the actual GO annotation, the annotation Cutoff was set to 55, and the GO Weight was set to 5. Terms were filtered by the Phaeophyceae database. $1.0*10^{-8}$ was used as the lower threshold for E-value, and HSP-Hit coverage Cutoff was set at 75. Hit filter was set to 500. Evidence codes were chosen according to the default options in the program, and an overview of the actual values can be found in the appendix.

After annotation, tables were exported from Omicsbox containing ids, ids, bitscore, evalue, similarity score, GO ids, count, GO terms, and category (Biological Process, Cellular Compartment, or Molecular Function). Rows containing no GO ids were removed (drop_na function). In R, this table was joined with data from earlier to include P values (left_join function), which were then -log10 transformed to use as scores for the GO terms. Because some GO terms contained very long descriptions behind commas, GO term strings were transformed to exclude commas and everything after (str_detect and strsplit functions). Subsequently, all rows were grouped by GO terms and mean -log10(p) values for each GO term were put into a new column, and the data ungrouped (group_by, summarize, and ungroup functions). Then, only unique GO terms were kept (the rest was discarded). The resulting data was grouped by category and the top 10 GO terms in each category was kept (distinct, group_by, slice_max, ungroup functions). Plots were made for each contrast using data made according to the method described above.

Results

Study 1: mineral composition in *S. latissima* individuals

Subset 1: rope section, thallus part, and bryozoa

An overview of the data in Subset 1 is shown in Table 5, including mean values for element content and Bryozoa score as well as mineral content and Bryozoa score for each rope section. Mean values for most elements appears to be higher in bottom samples than in the stipe, but bromine is more concentrated in the stipe. The differences were indeed mostly significant (except for iodine content) according to two-way ANOVA tests (Table 6), and are illustrated more clearly in Figure 4.

Table 5: Mean values for Subset 1 (10 individuals with both a stipe and bottom sample present) for 8 elements and Bryozoa score. Values are separated for stipe and bottom samples, as well as for the different rope sections (depth). Rope section 5 is the deepest part of the rope, while 8 is the highest up. Element abbreviations: I: iodine, As: arsenic, Br: bromine, P: phosphorus, V: vanadium, Co: cobalt, Cd: cadmium, Pb: lead. Element values are in mg kg^{-1}. Bryozoa values are based on a scoring system from 1 to 4 where 1 is the least amount of bryozoa and 4 is the most.

	Stipe					*Bottom*				
Rope section	*5*	*6*	*7*	*8*	*Mean*	*5*	*6*	*7*	*8*	*Mean*
I	5267	4767	4550	4700	4860	5367	5967	4200	5050	5250
Br	1900	1500	1450	1300	1570	1267	1200	1045	725	1094
As	66.7	69.7	70.5	60.0	67.0	97.3	103.3	80.0	86.0	93.4
Cd	1.43	1.13	0.97	0.81	1.12	2.57	2.27	1.45	1.50	2.04
Pb	0.153	0.107	0.050	0.047	0.097	0.283	0.263	0.210	0.087	0.223

V	0.34	0.31	0.29	0.28	0.31	1.23	1.04	1.25	0.69	1.07
P	2133	1800	1950	1500	1870	2633	2267	2250	1850	2290
Co	0.060	0.048	0.060	0.052	0.055	0.117	0.108	0.130	0.072	0.108
Bryozoa	1.3	2.0	1.0	2.0	1.6	2.0	2.3	3.0	2.0	2.3

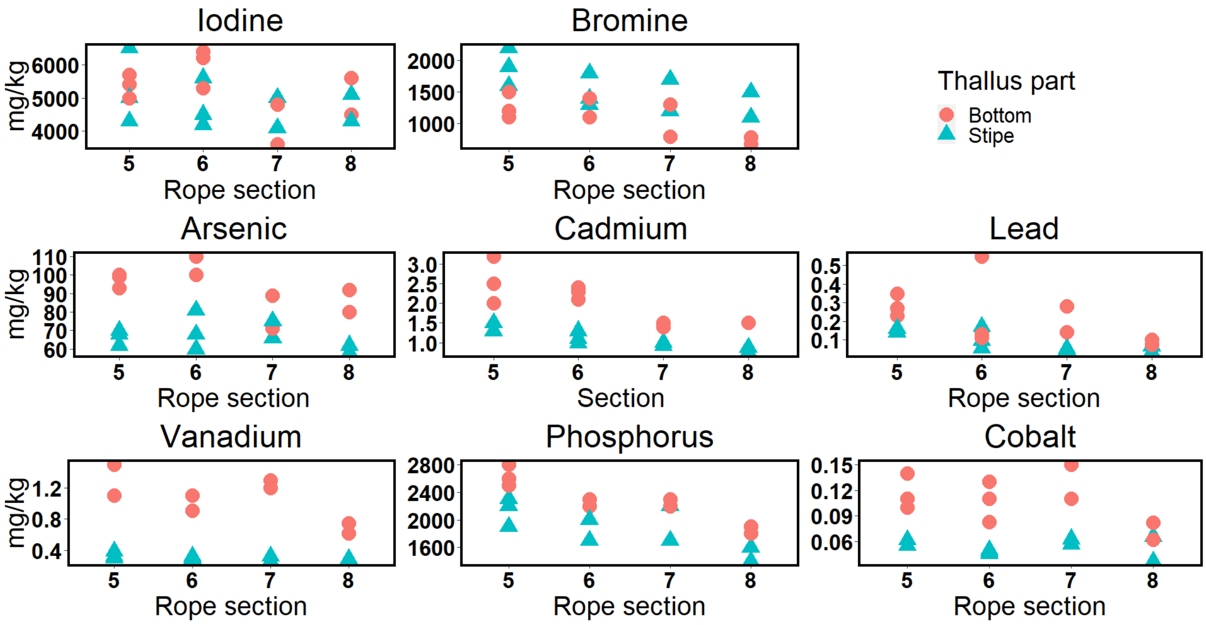


Figure 4: Scatterplots for 8 inorganic elements as functions of rope section in S. latissima. *Data is from Subset 1. Samples from the stipe are orange spheres while the samples from the bottom of the blade are aquamarine triangles. All y-axes are in mg/kg, although the scale is different for each subplot.*

Table 6: Results from a 2-way main effects ANOVA model looking at 8 elements as functions of rope section and thallus part. Data is from Subset 1. The numbers stated are p-values with significance levels: p = 0,05, p = 0,01**, p = 0,001***, - = no significance. Element abbreviations: I = iodine, As = arsenic, Br = bromine, P = phosphorus, V = vanadium, Co = cobalt, Cd = cadmium, Pb = lead.*

	I	*Br*	*As*	*Cd*	*Pb*	*V*	*P*	*Co*
Rope section	-	0.019 *	-	0.0008 ***	-	0.03 *	6.6*10^{-5} ***	-
Thallus part	-	0.0006 ***	3.5*10^{-6} ***	3.5*10^{-6} ***	0.02 *	10-8 ***	3.2*10^{-5} ***	8.8*10^{-6} ***

Looking at Table ?, Bryozoa score seems to be higher (more colonization) for bottom samples when comparing rope sections. Bottom samples also have higher overall mean for Bryozoa score (2.3) than stipe samples (1.6). Using one-way ANOVA tests, no element was significantly affected by Bryozoa score at p = 0.05, but arsenic was affected at a significance level of p = 0.1.

From Figure 4, it becomes clear that bottom samples have higher content of As, Cd, Pb, V, P, and Co. For iodine content there is no clear tendency in either thallus part or rope section. Bromine can be seen to be more concentrated in the stipe than in the bottom. In addition, there is a tendency that Br, Cd, Pb, V, P, and Co are more concentrated in the lower numbered rope sections (deeper).

From the two-way main effects ANOVA (Table 6) it was confirmed that rope section significantly affected Br, Cd, V, and P, but not Pb. The effect of rope section is significant for lead at p = 0,1, but not at p = 0.05. As suspected from the scatterplots, thallus part dominated the variance in the dataset, and was significantly affecting all elements except for iodine.

Subset 2: rope section, ash content, and fresh weight

Because thallus part was such a driving factor of the variation in Subset 1, Subset 2 was used for further analyses which included only bottom samples. Rope section was investigated also in this dataset, as well as % ash content, and weight.

In Figure 5 is seen the revisiting of element content versus rope section, for Subset 2. In contrast to the result in Subset 1, iodine and arsenic differed significantly between rope sections in this dataset. However, the relationship does not seem to be linear. For both elements, concentration tend to increase from rope section 5 to 6, decrease in section 7, then increases slightly in section 8. Phosphorus and vanadium were also significantly affected by rope section, and the correlation is negative. Br and Cd were not significantly affected by rope section, even though they were so in Subset 1. One sample seems to be an outlier in terms of Pb concentration containing more than 0.5 mg/kg.

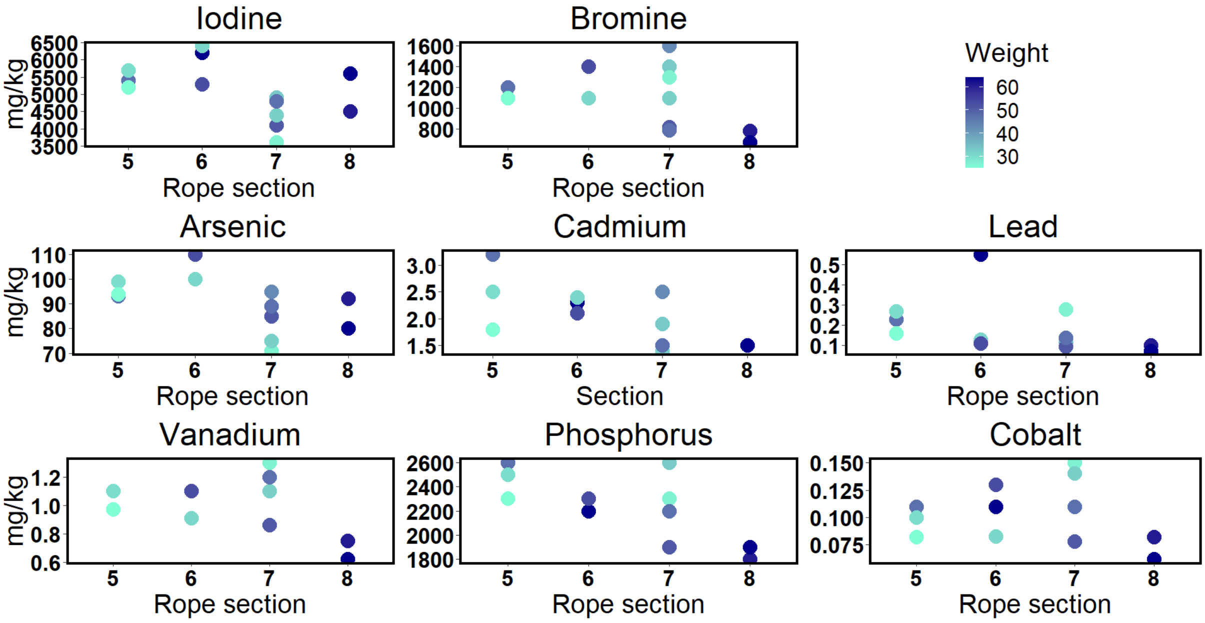


Figure 5: Scatterplots for 8 inorganic elements as functions of rope section in S. latissima. *Data is from Subset 2. Colour indicates thallus size (wet weight in g) ranging from the smallest thalli shown in aquamarine and the biggest thalli shown in dark blue. All y-axes are in mg/kg, although the scale is different for each subplot.*

Figure 6 show elements plotted against % ash content. For most elements there seems to be little variation associated with ash content, although there is a slight tendency that element contents are increasing with increased ash content. For Br and P this positive correlation was significant at p = 0.01.

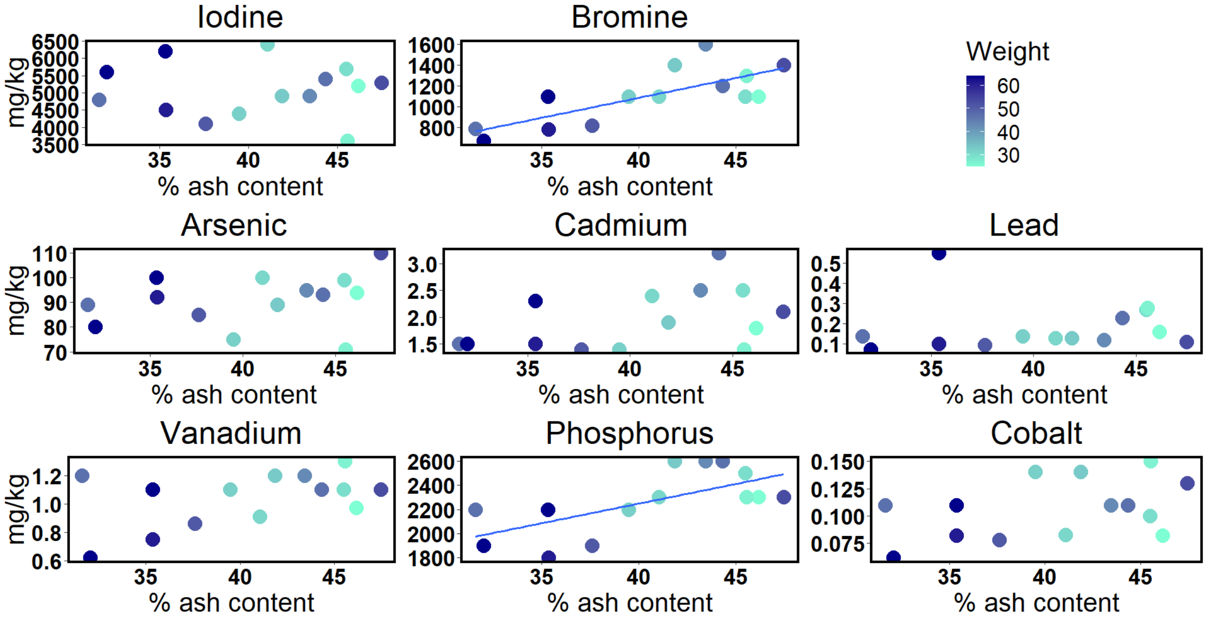


Figure 6: Scatterplots for 8 inorganic elements as functions of % ash content in S. latissima. *Data is from Subset 2. Colouring and shape of the coordinates represents wet weight of the individual the sample originates from. All y-axes are in mg/kg, although the scale is different for each subplot.*

Regarding thallus size (measured wet weight of individuals), Figure 7 indicates that there is a negative relationship between thallus size and ash content. However, it is difficult to know which variable affects the other (which is the predictor, and which is the response). Weight was also plotted against rope section (Figure 7), in which thalli growing deeper seemed to be smaller, but the relationship was only significant at p = 0.1.

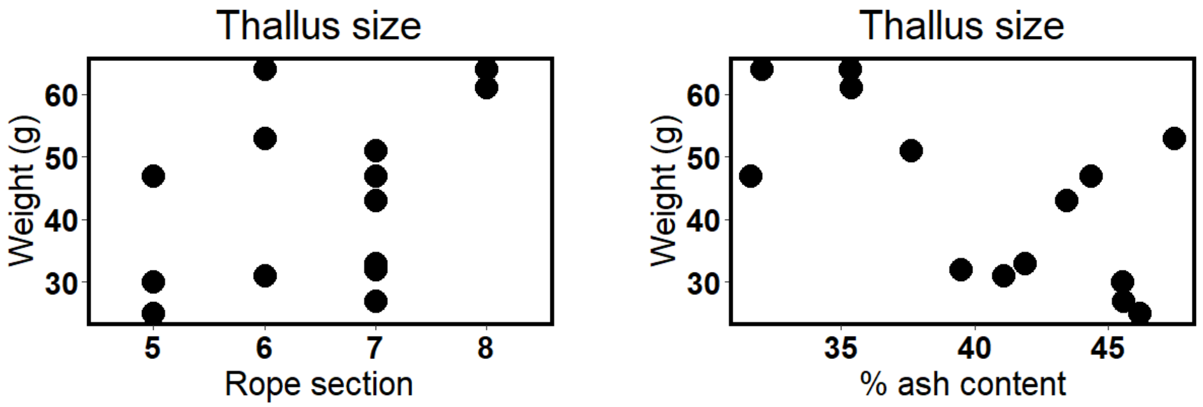


Figure 7: Fresh weight (g) for S. latissima *individuals plotted against rope section and % ash content in Subset 2 (only samples from the bottom of the blade, n = 14).*

Study 2: Transcriptome analysis of light-stressed *S. latissima* individuals

Analysis of mineral profile

The samples from the light-stress experiment were analysed to check the levels of various elements (I, As, Br, P, V, Co, Cd, Pb). A preliminary Principal Components Analysis (PCA) was used to look for patterns in the data and to get an overview (Figure 8). Principal component 1 (PC1) explains 32.85 % of the variation and most of the elements are positively correlated with this axis, except cobalt. Samples from the 40 µmol light control (marked red) are mostly positively correlated with PC1, and samples treated with 100 or 250 µmol light are mostly negatively correlated. Most samples being treated with 100 or 250 µmol light for 9 days are in the -0.2 area of PC1, but no element vectors point clearly in this direction. For each light intensity including control, the day 9 samples are further to the left along PC1 than day 3 and day 1 samples. PC2 explains 24.8 % of the variance, but seems not to be associated with either time or light intensity. Vanadium is somewhat correlated with PC2, while cadmium, arsenic and to some degree cobalt and bromine are negatively correlated with PC2.

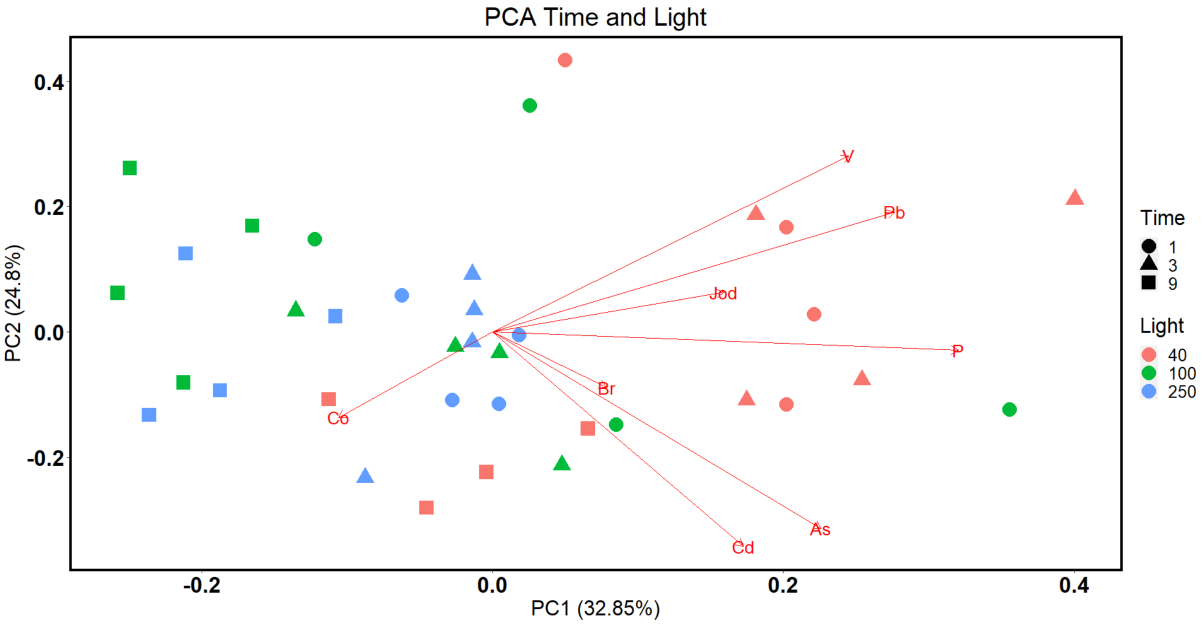


Figure 8: Principal component analysis (PCA) for 8 response variables (Elements: I, As, Br, P, V, Co, Cd, Pb). The dataset contains samples from a light-stress experiment, where

samples were exposed to 40, 100 or 250 µmol light for either 1, 3 or 9 days. Colours indicate light intensities and different shapes indicate duration (days) of stress.

Scatterplots were made for each element (Figure 9) as functions of time (days). Day 0 samples are from the end of the acclimation period and therefore only includes samples treated with 40 µmol light.

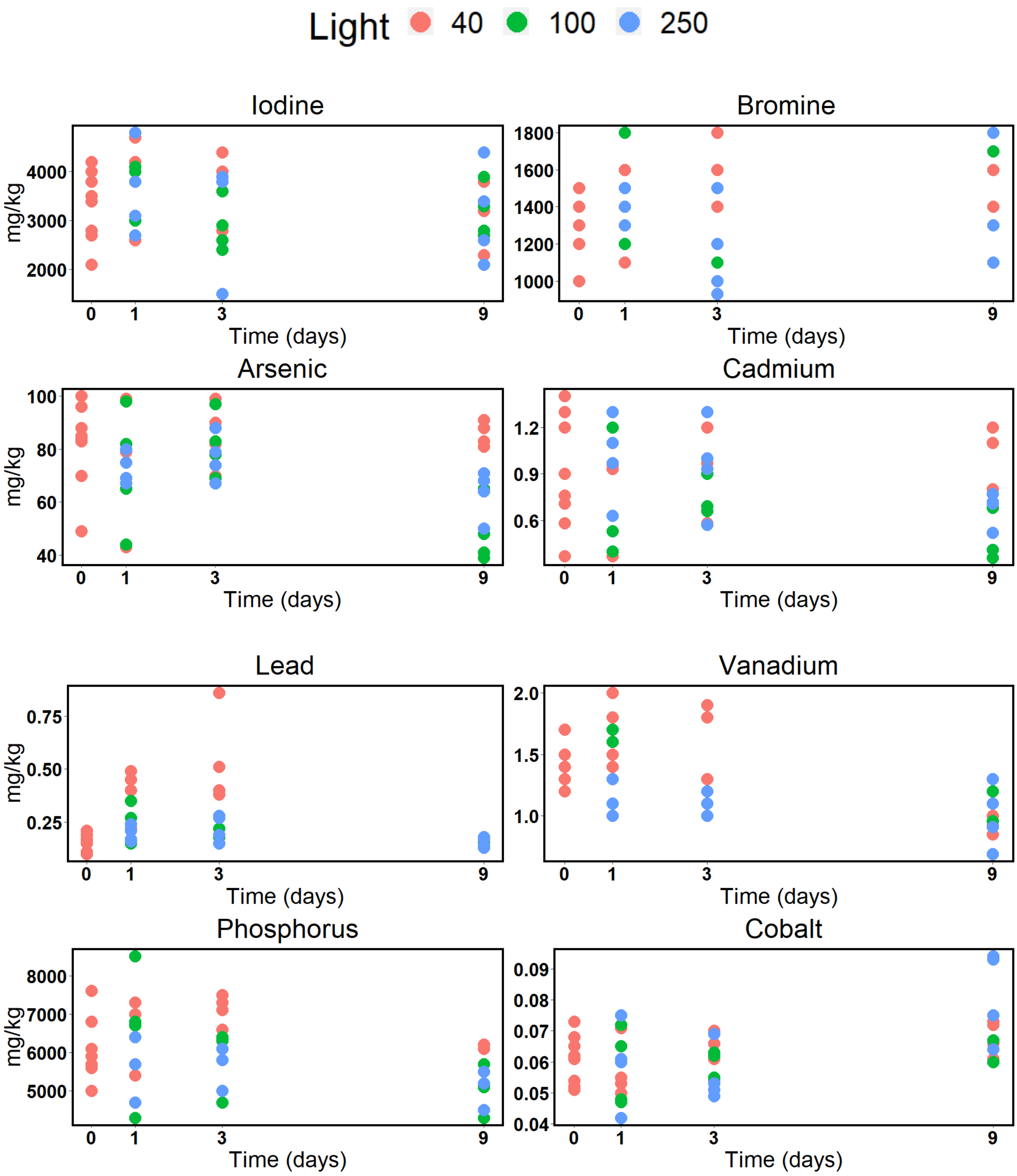


Figure 9: Scatterplots for iodine, arsenic, bromine, phosphorus, vanadium, cobalt, cadmium, and lead as functions of time (days) in S. latissima*. Colouring of the coordinates indicate the intensity of light-stress that they were exposed to during treatment. Day 0 samples were frozen in liquid nitrogen at the end of an acclimation period and were exposed to 40 µmol light.*

A two-way ANOVA was used to check the effects of time and light stress as well as interaction on the chemical profile in *S. latissima* (Equation 7). This analysis excludes day 0 control samples, although they are included in the scatterplots in Figure 9.

Levels of vanadium, phosphorus and to a certain degree arsenic tend to decrease with time. Especially on day 9 of treatment concentrations were noticeably lower. From the ANOVA it was confirmed that these elements were significantly affected by both time and light (Table ?). The effect of light is easily observable for phosphorus in the scatterplots, and for arsenic levels in day 9 samples. Vanadium content at days 1 and 3 was clearly lower for samples treated with high light (250 µmol light).

Looking at the scatterplots (Figure 9), cobalt was the only element to increase steadily over time. This relationship was found to be significant, but cobalt levels were not significantly affected by light intensities. Lead content also shows an interesting pattern, increasing a lot during the first days of treatment (days 1-3), especially when only exposed to 40 µmol light. But after 9 days lead levels are back to pre-treatment levels. In the ANOVA both light and time was confirmed to have significant effect on lead levels and there was also interaction between light and time.

Iodine, bromine, and cadmium were the most stable elements across time. Iodine tended to increase on day 1 of treatment but then returned to acclimation period levels. In addition, there are no apparent patterns of light intensities for either iodine, bromine, or cadmium. In agreement with these observations, no significant effect of either time or light was found for any of these elements.

Table 7: Results from two-way ANOVA analysing the effect of time (days), light stress and interaction on chemical profile in S. latissima*. Dataset includes all samples from treatments and no samples from the end of the acclimation period (day 0). The numbers stated are F values. Stars are based on p-values with significance levels: p = 0,05*, p = 0,01**, p = 0,001***, - = no significance. Element abbreviations: I = iodine, As = arsenic, Br = bromine, P = phosphorus, V = vanadium, Co = cobalt, Cd = cadmium, Pb = lead.*

	I	*As*	*Br*	*P*	*V*	*Co*	*Cd*	*Pb*
Time	-	F: 3.8 *	-	F: 5.3 *	F: 9.4 ***	F: 8.8 **	-	F: 10.8 ***
Light	-	F: 3.7 *	-	F: 5.6 **	F: 5.9 ***	-	-	F: 12.0 ***
Interaction time*light	-	-	-	-	-	-	-	F: 4.1 *

Control

A one-way ANOVA was used to check whether cutting the tissue had any effect on chemical profile in *S. latissima* (Equation 5). The samples used in this analysis were from the end of the acclimation period and had only been exposed to 40 µmol light. Only for lead content was there a significant difference between cut and uncut tissue. Another one-way ANOVA was used to check the effect of time on the chemical profile (Equation 6), and only samples exposed to 40 µmol light were used here as well. For vanadium and lead contents there was a significant effect of time.

RNA-seq results

Quality control, de novo assembly and alignment

Table ? contains stats from the Trinity de novo assembly. The total number of genes excluding isoforms outputted by Trinity is marked as Total trinity "genes", while Total trinity transcripts includes alternatively spliced isoforms. According to these stats, the median and average contig lengths were longer among all transcripts than among only the "genes" (only the longest isoform per gene).

Table 8: General statistics related to the de novo assembly. Units are noted in the Table, except Total trinity "genes" which is the total number of transcripts excluding isoforms in the finished assembly, and Total trinity transcripts is the total number of transcripts including isoforms. Percent GC: Guanine-cytosine content divided by all bases.

General	
Total trinity "genes"	414164
Total trinity transcripts	552394
Percent GC	52.92 %
Stats based on all transcript contigs	
Median contig length	349 bp
Average contig	614.95 bp
Total assembled bases	339.7 Mbp
Stats based on only the longest isoform per "gene"	
Median contig length	312 bp
Average contig	507.83 bp
Total assembled bases	2100.3 Mbp

94 % of the BUSCO groups searched were complete, although only 3 % were single-copy. 91 % were complete and duplicated.

Table 9: BUSCO output showing the completeness of the de novo assembly. BUSCOs represent markers that were either found (complete or fragmented) or not found in a database (stramenopiles).

Results from dataset stramenopiles

94 Complete BUSCOs (94%)	
3 Complete and single-copy BUSCOs	91 Complete and duplicated BUSCOs
5 Fragmented BUSCOs	
1 Missing BUSCO	
100 Total BUSCO groups searched	

The estimated variance in gene abundance across all samples and replicates was calculated to be ≈ 0.49 (BCV). After stabilizing the variance between counts based on the BCV, the top 500 most variable genes (based on read counts between all samples) were selected and analysed using a principal component analysis which is shown in Figure 10. The first component (PC1) explains 43.8% of the variation and differentiates samples quite clearly by light intensity along its whole axis. Firstly, the two types of day 0 samples were exposed to 40 µmol light but there seems to be a difference between along PC1. Continuing along PC1 were the rest of the samples exposed to 40 µmol light, then the samples exposed to 100 µmol light, and lastly the samples exposed to 250 µmol light. PC2, on the other hand, only explained 17.3% of the variation, but there were some level of differentiation in terms of duration of treatment along this axis. For MIN samples, D9 samples are more positively correlated with PC2. For both MED and MAX samples, D9 samples are more negatively correlated with PC2, and D1 samples are more positively correlated.

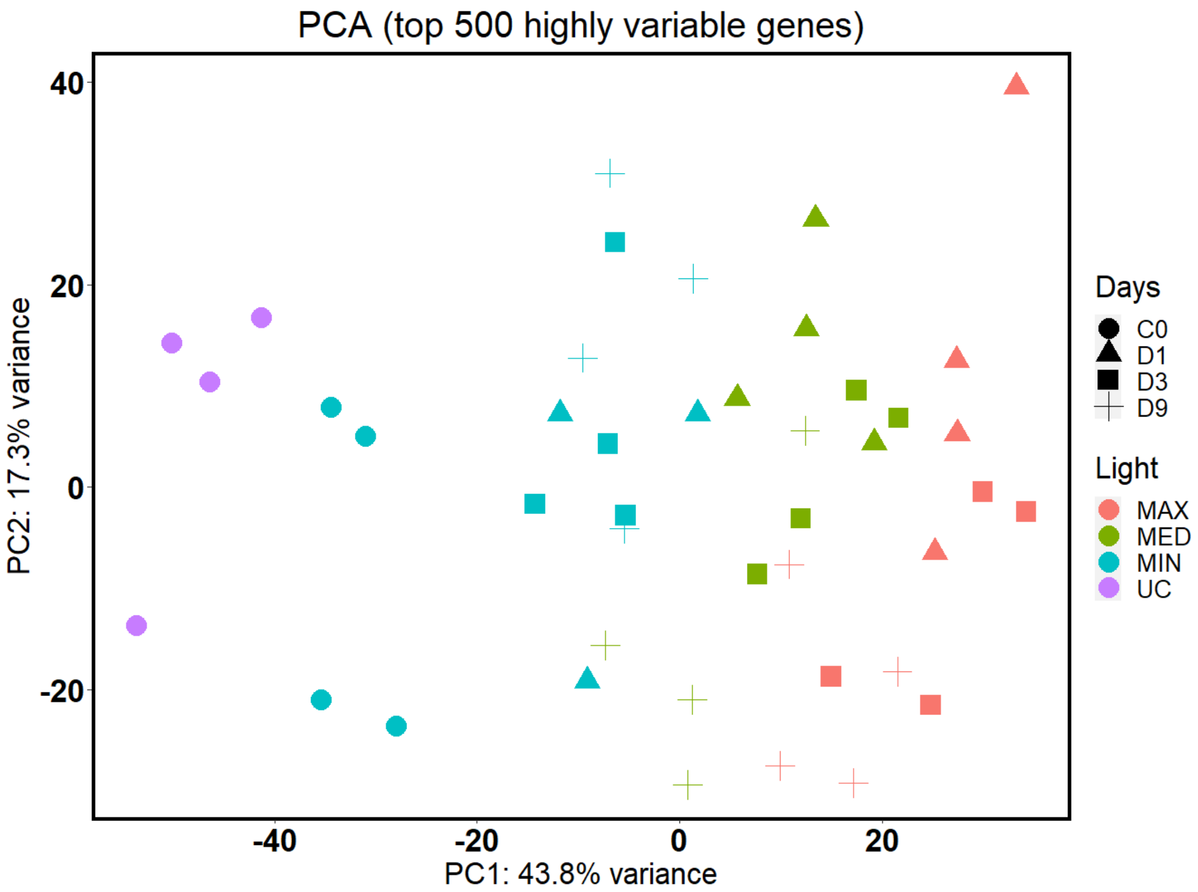


Figure 10: Principal component analysis (PCA) for the 500 most variable genes among the transcripts that aligned to the assembly. Different colors represent light intensities while shapes represent treatment duration in days.

Figure 11 shows Mean-Difference plots visualizing differentially expressed genes (DEGs) compared to average log counts (CPM). Genes had to be up or down regulated by more than $\log_2(1.5) \approx 58\%$, using an adjusted p-value of 0.05 (FDR), to be considered differentially expressed. All of the chosen contrasts (Table 4) are shown, in addition to MED1 vs C1 (samples exposed to 100 µmol light for 3 days vs day 3 control samples) and MED9 vs C9 (samples exposed to 100 µmol light for 9 days vs day 9 control samples). These two contrasts had very few DEGs. MAX1 vs C1, MED9 vs MED1, and MAX9 vs MAX1, are the most apparent contrasts that contain a lot of DEGs, and to a lesser extent also MAX3vsC3 and MAX9vsC9. There is a fair amount of DEGs in the time control (C9 vs C0), especially upregulated genes (438 upregulated and 151 downregulated). The cut control (C0 vs UC) contained a small amount of DEGs, but still more DEGs than MED1 vs C1 and MED9 vs C9.

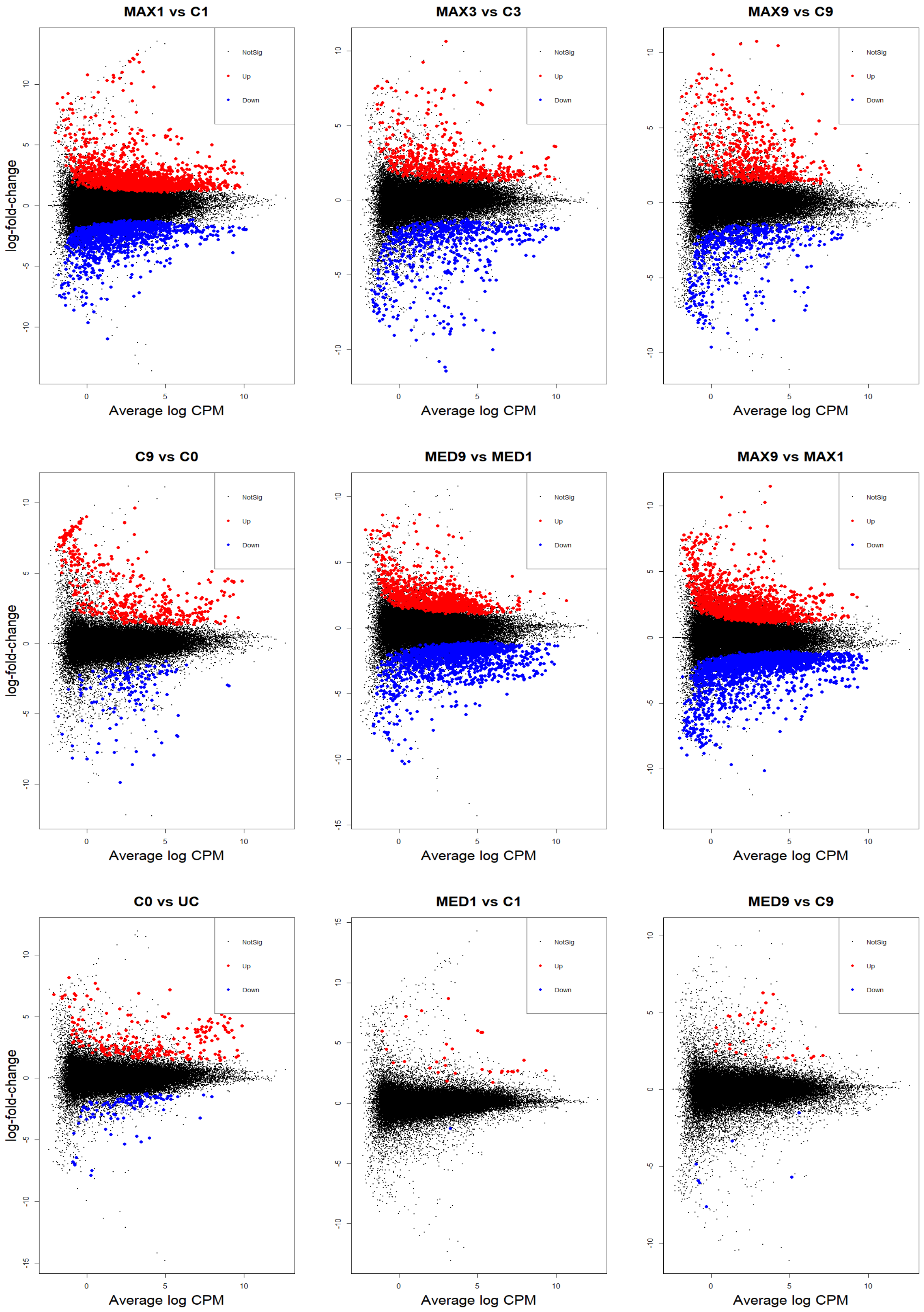


Figure 11: Mean-Difference plots showing the log fold change in gene expression compared to the control plotted against average log CPM plotted on the x axis. Red and blue dots are

genes that are up or down regulated respectively by more than 0.58 %. 9 contrasts: MAX1vsC1, MAX3vsC3, MAX9vsC9, C9vsC0, MED9vsMED1, MAX9vsMAX1, C0vsUC, MED1vsC1, MED9vsC9.

Figure 12 shows an upset plot visualizing the intersection of regulated genes between the treatment contrasts. The top 3 groups are set apart from the rest in terms of the number of differentially expressed genes. The effect of time at high light (MAX9-MAX1) caused up -or down regulation of 3641 genes in total. MED9-MED1 and MAX1-C1 caused regulation of in total 2700 and 2824 genes respectively. Out of the genes regulated in the top 3 groups, many genes were only regulated in that group. 1535, 1444, and 1417 genes were only regulated in MAX9-MAX1, MED9-MED1, and MAX1-C1 respectively. There was notable overlap between MAX9-MAX1 and MED9-MED1 (771 genes) as well as between MAX9-MAX1 and MAX1-C1 (598 genes). 468 genes were only regulated in MAX3vsC3 and 373 genes were only regulated in MAX9vsC9 among the contrasts investigated. MAX3vsC3 and MAX9vsC9 had 111 DEGs in common.

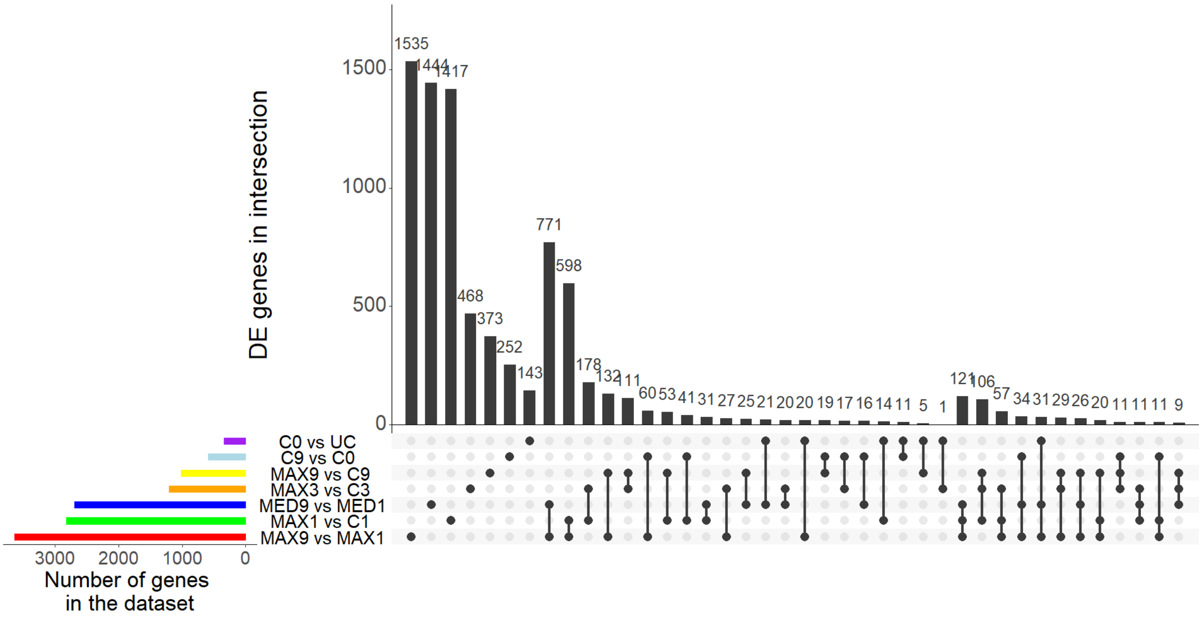


Figure 12: Upset plot for differentially expressed genes in 7 comparisons (MAX9vsMAX1, MED9vsMED1, MAX1vsC1, MAX3vsC3, MAX9vsC9, C9vsC0, and C0vsUC. The dot section shows by lines which groups have intersecting DEGs, and the black lines show how many genes intersect (dots with no line represent genes that are uniquely regulated in that group. Coloured bars represent the total amount of DEGs (intersecting and unique) for each group.

The upset plot does not say anything about which genes are up or down regulated in different treatments, and although the number of DEGs does say something about the magnitude of the total response to various treatments, it is not a useful parameter for understanding the processes that underlie these responses. A high number of DEGs does not necessarily indicate interesting genes. In the downstream analyses I therefore focused on the light contrasts (MAX1vsC1, MAX3vsC3, and MAX9vsC9), which are the most interesting contrasts a priori. Figures 13-15 show the top Gene Ontology terms (GO terms) for MAX1vsC1, MAX3vsC3, and MAX9vsC9. GO terms are sorted into biological process (BP), cellular component (CC) and molecular function (MF), distinguishing between different levels of organisation. In all 3 contrasts, GO terms in the CC category had lower scores than BP and MF, measured in -log10(p-value).

Among the top GO terms for MAX1vsC1, "xanthophyll cycle" and "violaxanthin de-epoxidase activity" both has high scores. Processes and activity involving arginine, malate, and glutamate is going on, as well as activity and transport of ATP and ADP. As can be seen among CC terms, higher organisation processes are also in motion, especially relating to the nucleolus. Notably, there is also peroxisome activity.

For DEGs in MAX3vsC3, the "nucleolus" term can no longer be seen among CC terms. Higher organisation processes now include activity in ribosomes, cytoplasm, membranes, and in photosystem II. The xanthophyll cycle appears to active for this contrast as well. Also, genes relating to O-acyltransferase activity, monooxygenase activity, and glutaminase activity appears to be differentially expressed.

Among the top GO terms for MAX9vsMAX1, several terms are associated to the same processes as in MAX3vsC3, such as cytoplasm, photosystem II, membrane, glutaminase, and ribosome activity. In addition, processes involving cellular oxidant detoxification and glutathione metabolism appears to be active.

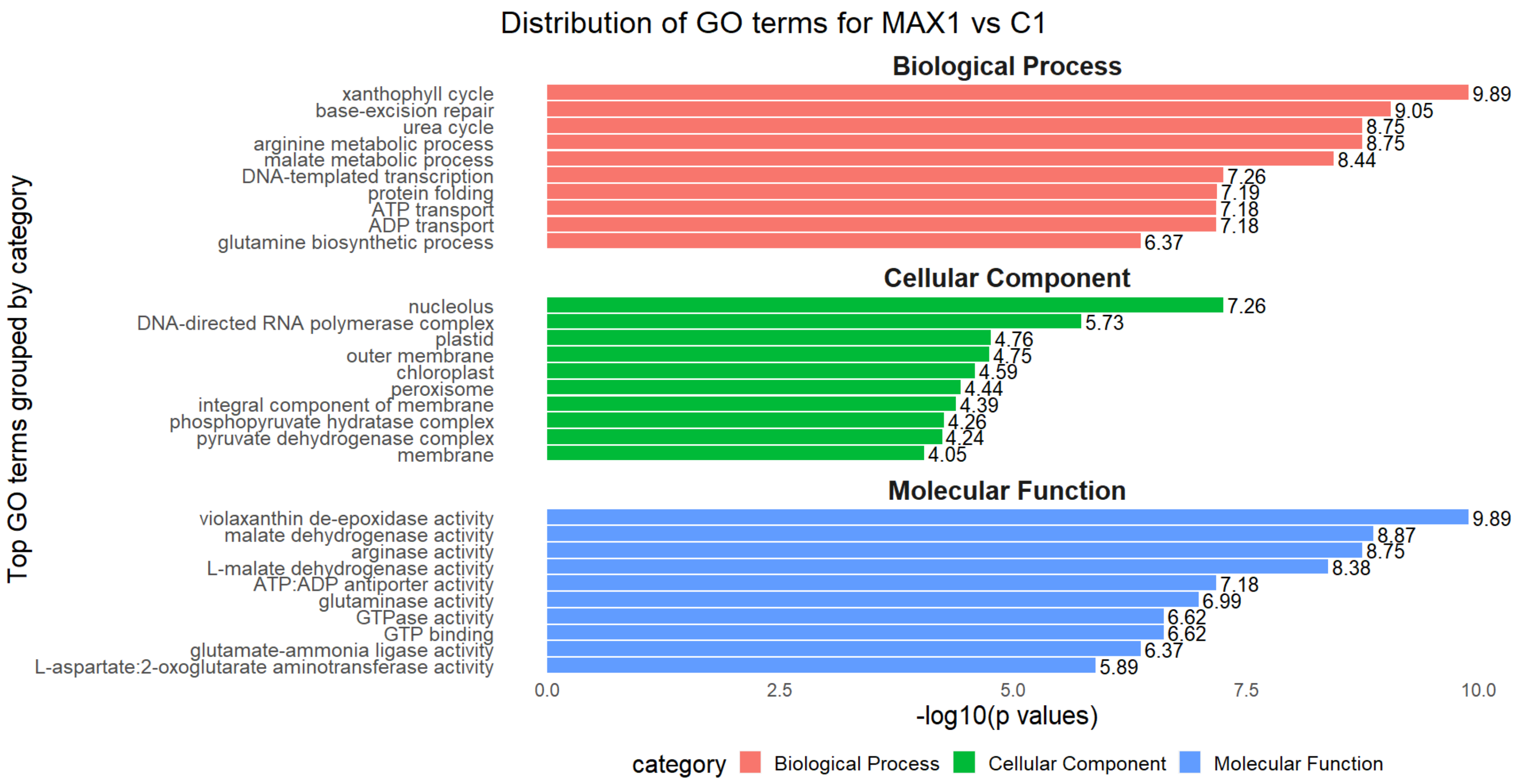


Figure 13: Plot showing the distribution of the top 10 gene ontology terms for MAX1-C1 in each category (Biological Process, Cellular Component, or Molecular Function). Each category is ordered by mean -log10 transformed p-values (-log10(p)), for each GO term.

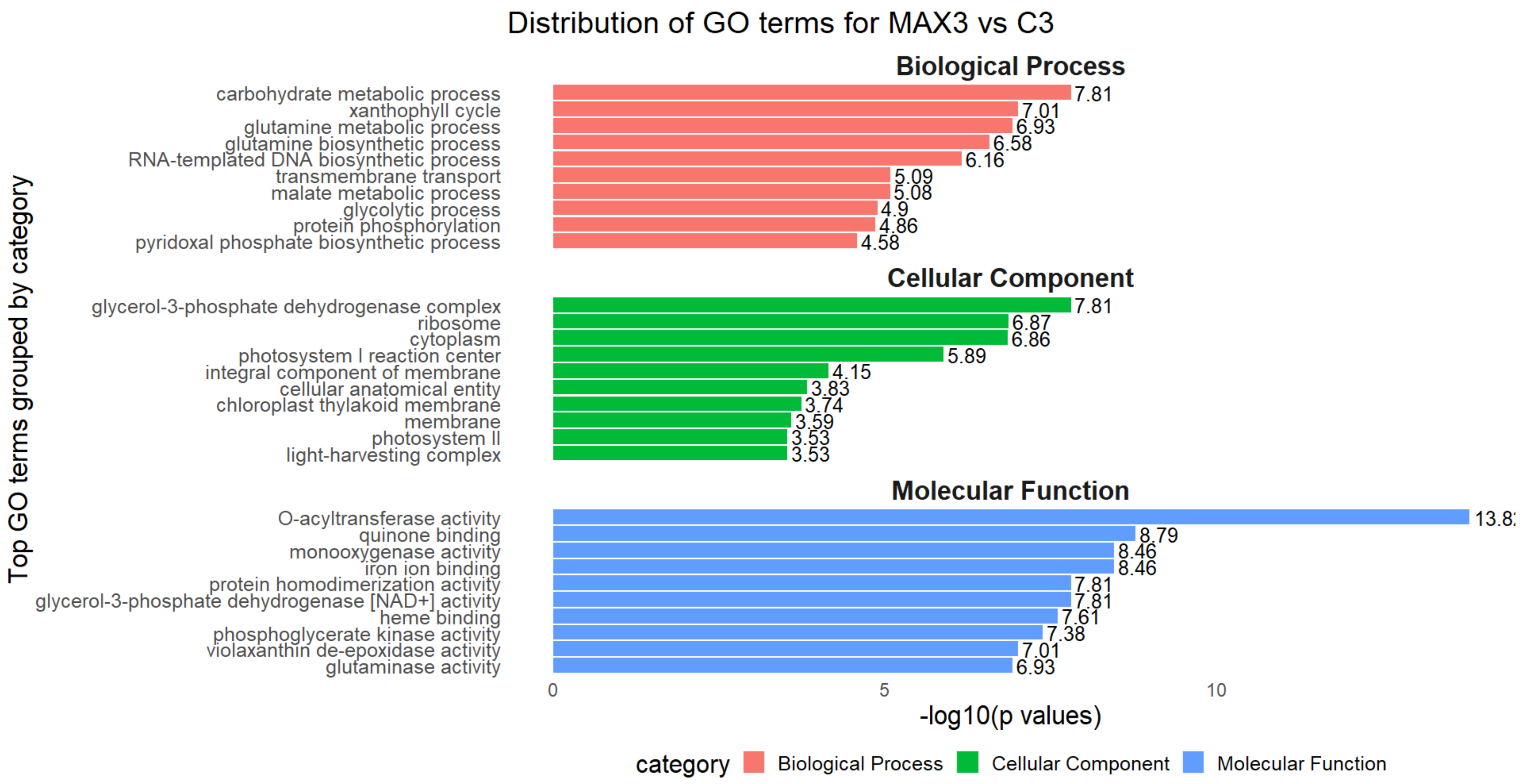


Figure 14: Plot showing the distribution of the top 10 gene ontology terms for MAX3vsC3 in each category (Biological Process, Cellular Component, or Molecular Function). Each category is ordered by mean -log10 transformed p-values (-log10(p)), for each GO term.

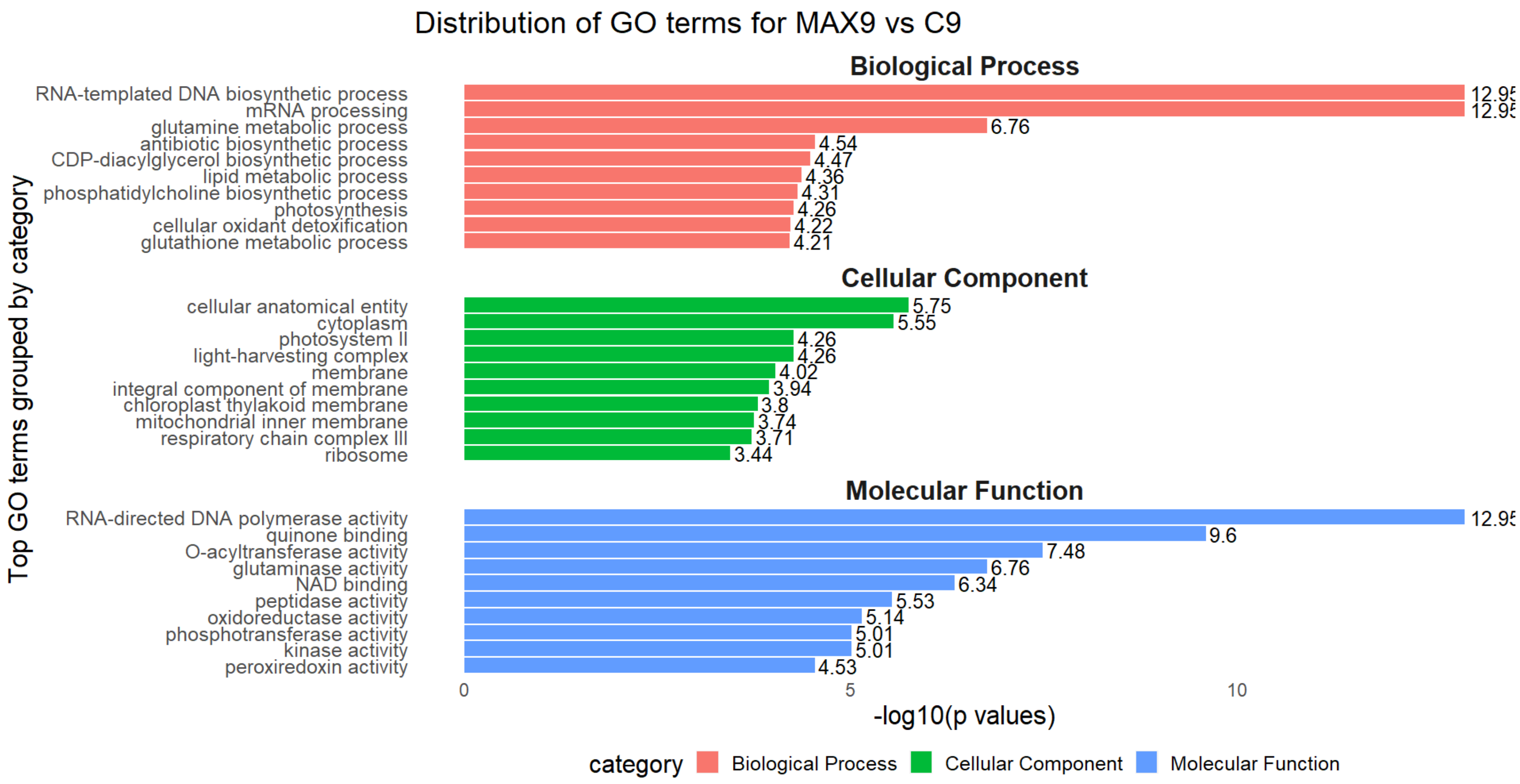


Figure 15: Plot showing the distribution of the top 10 gene ontology terms for MAX3vsC3 in each category (Biological Process, Cellular Component, or Molecular Function). Each category is ordered by mean -log10 transformed p-values (-log10(p)), for each GO term.

To look closer at the effect of light on gene products and specifically proteins related to vHPOs, the glutathione, and the xanthophyll cycle, heatmaps were made comparing relative expression of annotated gene products across samples treated with 250 µmol light as well as the controls. In Figure 16 is shown the top 10 most variable genes overall, while in Figure 17 is shown relative expression for selected genes thought to be associated to vHPOs and heavy metals.

For the top 10 most variable genes, there is a clear difference between treatment and controls samples for especially 5-methyltetrahydropteroyltriglutamate--homocysteine methyltransferase and HSP domain-containing protein. Glucose regulated protein /BiP is upregulated only on the first day of 250 µmol light and downregulated in the other treatments. CMD domain-containing protein is upregulated in all treatment groups versus the control groups. A putative lipoprotein gene, on the other hand, was upregulated in all the control groups compared to the treatment groups. The same was to a lesser extent also true for a Glycerol-3-phosphate dehydrogenase. EsV-1-12 was clearly upregulated in all groups compared to MAX1, and a Peptidase S74 domain-containing protein was clearly upregulated in all groups compared to C1. Tubulin alpha-2 chain was upregulated in MAX9, C3, and C9, while a Rieske domain-containing protein was downregulated in MAX1 and C1.

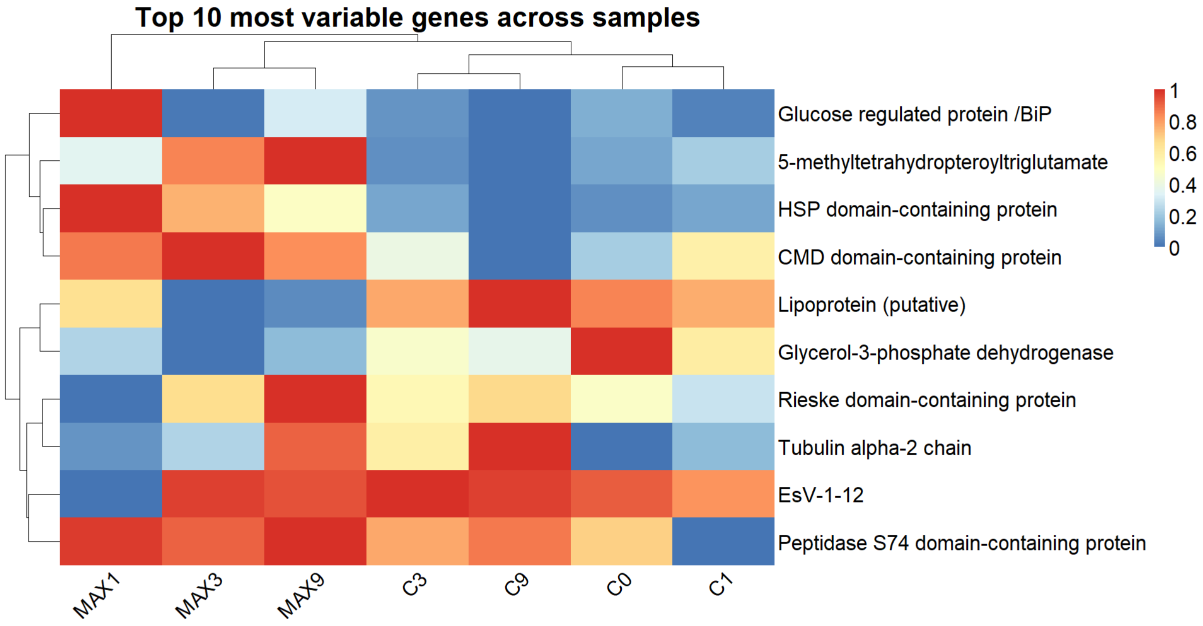


Figure 16: Heatmap showing the top 10 most varibale gene products (proteins) across 7 sample groups (C0, C1, C3, C9, MAX1, MAX3, and MAX9) in S. latissima. *The gene products were selected by ordering all gene products by variation in logCPM across the samples mentioned above, selecting the top 10 proteins, and then range-normalizing the resulting list*

from 0 to 1. The control groups are samples exposed to 40 µmol m^{-2}s^{-1} light for 0, 1, 3, and 9 days (C0, C1, C3, C9 respectively), and the treatments groups are samples exposed to 250 µmol m^{-2}s^{-1} light for 1, 3, and 9 days (MAX1, MAX3, and MAX9 respectively). All groups had 4 replicates, except for C1 which had 3. Protein annotations were fetched using diamond blastx from the phaephyceae database.

Regarding the vHPOs, vBPO seems to be strongly upregulated in MAX9, and to a certain extent in MAX3. vIPO also tend to be upregulated in the treatment groups, especially in MAX1. Out of the various glutathione S-transferase gene products, there seems to be higher expression in treatment groups, as well as in C1. Glutathione reductase has higher expression in treatment groups compared to controls. A chloroplastic violaxanthin de-epoxidase gene product was upregulated strongly in MAX1 and MAX3, and to a lesser degree in C1 and C3.

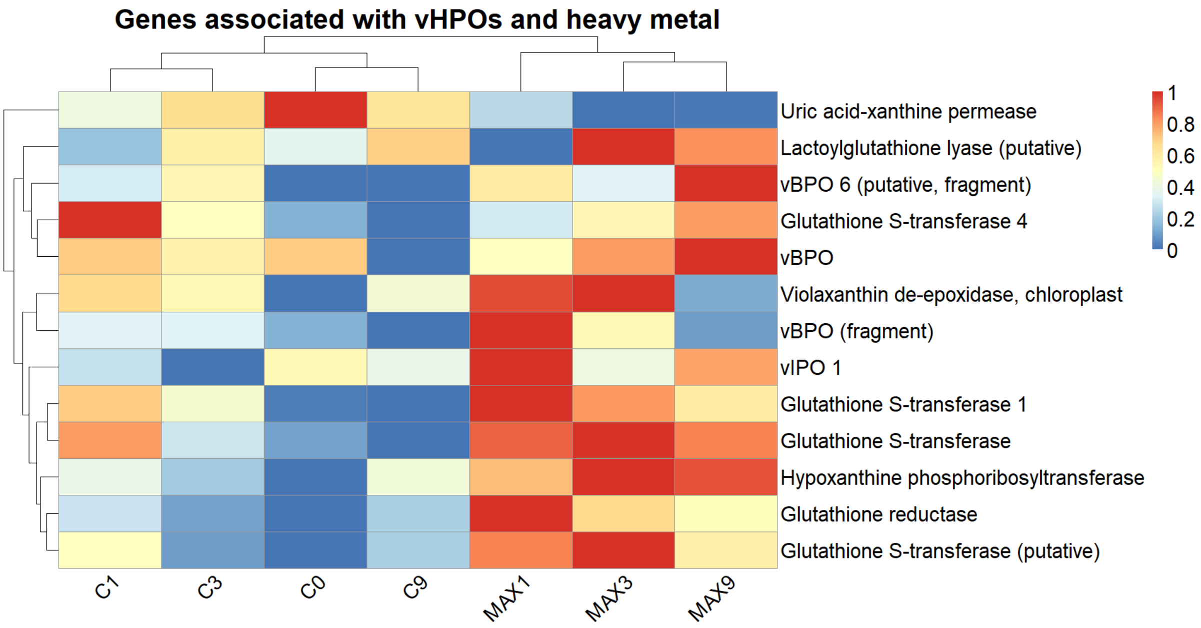


Figure 17: Heatmap showing range-normalized counts (logCPM) of selected gene products across 7 sample groups (C0, C1, C3, C9, MAX1, MAX3, and MAX9) in S. latissima. *The gene products were selected based on a manual search among all annotated gene products for the keywords "vanadium", "glutathione", and "xanthin". The control groups are samples exposed to 40 µmol m^{-2}s^{-1} light for 0, 1, 3, and 9 days (C0, C1, C3, C9 respectively), and the treatments groups are samples exposed to 250 µmol m^{-2}s^{-1} light for 1, 3, and 9 days (MAX1, MAX3, and MAX9 respectively). All groups had 4 replicates, except for C1 which had 3. Protein annotations were fetched using diamond blastx from the phaephyceae database.*

Discussion

Study 1: mineral composition in *S. latissima* individuals

Subset 1

Bryozoa

The discovery of serious bryozoan fouling of much of the sample tissue in Study 1 posed some challenges. The bryozoan *Membranipora membranacea* is an epiphyte that colonize seaweeds in temperate waters and has been found to contain high levels arsenic (Getachew et al., 2015). The same study found colonized *S. japonica* tissue to have elevated levels of iodine and decreased levels of cadmium after removing the bryozoans. For this reason, only some of the samples were analysed for inorganic elements. Samples from the stipe and bottom of the blade had relatively few colonies and were therefore chosen for further analysis. The mean Bryozoan score in bottom samples were 2.3 and 1.6 in stipe samples. Bryozoan score had a significant effect on As at p = 0.1. This is not a convincing result, especially because the scoring system was visual. However, bryozoans are known to have high contents of As (Getachew et al., 2015), so the result is hardly surprising.

Thallus part

There were no significant differences in iodine content between stipe and the bottom of the blade. For iodine it would have been interesting to see the comparison to middle and tip samples, because these older parts of the thallus are believed to contain less iodine than the stipe and meristematic region (Amat & Srivastava, 1985; Roleda et al., 2018). Unfortunately, these were not analysed because of Bryozoan fouling as mentioned. At the same time, most iodine is found in the peripheral tissue (Küpper et al., 2013; Verhaeghe et al., 2008), and it was not a goal of this study to investigate cross-sectional variation.

On average 1570 mg/kg bromine was found in stipe samples, while bottom samples contained an average of 1094 mg/kg. (Sharma et al., 2018) found bromine values in cultivated *S. latissima* harvested in June to be on average between 1.2 and 1.4 g/kg, which is comparable to the values found in this study. (Verhaeghe et al., 2008) found bromine levels in the stipe to be on average 1.51 +- 0.02 mg/g, which is also in agreement with the levels found here. It is not clear why bromine content is higher in the stipe, while iodine is about the same in stipe and blade. It has been shown that vBPO plays an important role in the adhesive system of the holdfast in in *F. serrata*, by cross-linking phenolic polymers (Berglin et al., 2004). This could be a partial explanation why more bromine is found in the stipe (or rather the holdfast) than the blade, if indeed vBPO and bromine is connected to the adhesive system in *S. latissima* as well.

Mean arsenic concentration is 67 mg/kg and 93.4 mg/kg in stipe and bottom samples respectively. These are levels of total arsenic however, and the inorganic fraction is suspected to be much lower. Roleda et al. (2019), computed average proportions of inorganic arsenic based on data from Almela et al. (2006) and Díaz et al. (2012). They found mean inorganic arsenic in Laminariales to be 1.72% of the total arsenic based on this. If we assume that the proportion of inorganic arsenic is about the same in these samples, the mean i-As would be about 1.15 mg/kg in stipe samples and 1.61 mg/kg in bottom samples. These levels are within the 3 mg/kg French maximum of i-As levels in algae condiments (Petursdottir et al., 2015).

The mean cadmium content was 1.12 and 2.04 mg/kg in stipe and bottom samples respectively, which is within the EU maximum level of 3 mg/kg cadmium in food products from dried seaweed (No, 1881). Sharma (2018) found the cadmium level in June-harvested *S. latissima* to be 0.8-0.9 at 3 and 8 meters respectively.

One of the aims of this study was to investigate whether one part of the blade can be used to predict the content of the whole blade. Obviously, it would have been better to have samples from all thallus parts, but this was not possible because of the bryozoan fouling as explained. In addition, the sample size was quite small for this experiment, and the results should be interpreted with that in mind. Despite these matters, the differences between the stipe and the bottom of the blade were quite clear and was significant for all elements except for iodine. Based on this, one should be careful with trying to predict mineral content for thallus parts based on the content in other thallus parts. Even though iodine was not significantly different between the bottom and stipe, there is evidence from the literature that the tip contains less iodine as mentioned in the introduction (Amat & Srivastava, 1985).

Subset 2

After removing the variation associated with thallus part, various patterns emerged regarding rope section, ash content, and thallus size.

Rope section, ash content, and fresh weight

It is not unreasonable to assume that rope section as a factor is synonymous with depth, and depth is the most interesting aspect of rope section at least in this study. However, it should be mentioned that rope sections could differ from each other in other ways than having different depth. For instance, a rope could be eroded to a varying degree across its length or have varying degrees of smoothness and roughness across its surface and thus affect growing conditions. Furthermore, the exact depth of the rope sections is not known in this experiment, making the results somewhat difficult to compare to other studies. However, the relative differences in depth between the rope sections is taken as an acceptable measure of depth for the purposes of this discussion.

In the scatterplots for mineral contents versus rope sections for Subset 2, Iodine and arsenic have some interesting sine shaped curves suggesting that the relationship between these elements and depth is not linear. There was a significant relationship according to ANOVA, but this is more likely to be random noise resulting from small sample size. There is no reason to think that depth is correlated with I and As by a sine wave relationship.

P and V were both significantly correlated with rope section. Individuals growing deeper contained more P and V. Similarly, Sharma et al. (2018) found vanadium contents for *S. latissima* sampled in June to be slightly higher at 8 m than at 3 m, but there was no mention if the relationship was significant. On the other hand, the same study found phosphorus content for the same samples to be slightly higher at 3 m than at 8 m, oppositely of what was found in this experiment. This is probably not a notable disparity, firstly because the actual depths of each rope section in this experiment is unknown, and so not necessarily comparable to other studies. Secondly, phosphorus was found to be significantly correlated (positively) to ash content in this experiment, meaning that at least some of the variation observed for phosphorus can be attributed to ash content. The same was true for bromine, that is, increased levels of bromine correlated with increased ash %.

Regarding thallus size compared to rope sections and ash content, individuals growing on higher up rope sections tend to be bigger, but there was no significant correlation in this regard. It is logical that individuals growing deeper has less sunlight available and therefore gains less biomass. There appears to be a negative relationship between thallus size and ash content. But the challenge is to decide which of these is the predicting factor and which is the response.

Study 2: Transcriptome analysis of light-stressed *S. latissima* individuals

Remarks on methods

RNA-extraction

It proved difficult to extract pure RNA samples from the sample tissues. The 260/230 ratio measured by NanoDrop was repeatedly found to be lower than 1, while ideal values are 2-2.2 for RNA samples (Nanodrop, 1975). Although there are several possible contaminants absorbing light at 230 nm, phenols are likely candidates as algae are known to be rich in phenols (Nielsen et al., 2020). For future projects doing RNA-extraction on brown macroalgae I suggest experimenting with the centrifugation speeds and duration during the RPE cleaning step, as well as possibly repeating the step. These modifications though tended to decrease the concentration of the eluated RNA, but leaving water in the spin columns for 1 minute on the bench before final eluation centrifugation should help increase concentration.

De novo assembly

The reference genome assembled had a high grade of completion (94%), but only 3% of the complete gene markers were single-copy. This suggests that one or more gene (or whole genome) duplication events has occurred during the evolution of *S. latissima* after its departure from the rest of the stramenopiles. Liu et al. (2019) found a high number of orthogroups (group of genes deriving from a single ancestral gene) in the genome of *S. japonica*, which is also indicative of duplication events during the recent evolution of *Saccharina* species. However, some of the duplication found in the de novo assembly is probably due to isoforms, which had not been filtered at this point in the pipeline. Another BUSCO analysis should probably have been performed after filtering for isoforms. Similarly, all parameters of the MultiQC report were ok, except for sequence duplication levels (appendix).

Moreover, according to the Trinity stats, the median and average contig lengths were longer among all transcripts than among the longest isoform per gene, which sounds to me like a contradiction. If the set of the longest isoforms per gene is a subset of the total number of transcripts (including isoforms), the former should always have longer average and median length. This observation suggests some sort of technical error which is beyond my current knowledge of Trinity.

Peculiarly, only the first attempt at creating a de novo assembly succeeded even though there were several attempts to make new de novo assemblies. Various errors or bugs caused these attempts to fail and because of time concerns I had to move on (de novo assembly is time consuming). It would be interesting to go back to this step and experiment. Firstly, I would filter for isoforms as the results from MultiQC and BUSCO suggested. I would also perform the Kraken2 blast before assembly to filter out unwanted sequences from other organisms. If the de novo assembly is pure from unwanted sequences, there is no need to blast the transcript reads as was done here because the assembly will anyways not contain unwanted sequences from other organisms that the transcripts can align to.

Differentially expressed genes

Biological coefficient of variation: can we know if the variation is due to environment or genetics?

A BCV of between 0.2 and 0.4 is usually considered optimal, but values of up to 0.6 can be acceptable in for instance population studies. The BCV in this study was found to be 0.49 across all samples. Although this is on the higher end, BCV naturally depends on the origin and treatment of the samples. Samples from clones or from the same individual are expected to have a lower BCV, but the samples in this experiment are from different individuals and can be expected to have slightly higher BCV. The PCA for the top 500 most variable genes indicates that more variance stems from light intensity than from treatment duration, although it is difficult to tell from a PCA how these factors interact.

Mean-Difference plots

From the Mean-Difference plots it becomes clear that there is not much DE between the treatments exposed to medium light (100 µmol $m^{-2}s^{-1}$ light) for 1 or 9 days and the corresponding control for that day (MED1 vs C1 and MED9 vs C9). Yet, MED9 vs MED1 has a relatively high amount of DEGs, which indicates that 100 µmol light does not cause regulation of many genes in *S. latissima*. Notably, there is some level of variance coming from the time control and the cut control. The time control tells us that keeping tissue samples in beakers for 16 days (7 days acclimation + 9 days treatment) resulted in a fair amount of DE. Similarly, cutting samples caused some regulation of genes compared to acclimating whole thalli over a 7-day acclimation period. Regarding the samples exposed to 250 µmol light, they all had a substantial amount of DE compared to their respective time controls. However, the effect was greatest on day 1 (MAX1vsC1), as can also be seen in the upset plot. A possible explanation is that *S. latissima* responds to high light stress fairly quickly, initiating processes that lead to long-term protection, and then attempts to revert back to a baseline.

Upset plots

The contrasts containing the most DEGs are MAX9vsMAX1, MED9vsMED1, and MAX1vsC1 as can be seen in the upset plot. There are many genes among these contrasts that does not intersect with other contrasts (they are only regulated in one contrast). There are notably few DEGs overlapping between MAX1vsC1 and MED9vsMED1 (31 genes), considering that there are in total 2824 and 2700 DEGs respectively in these contrasts. This illustrates the difference between sudden, intense light stress, and moderate light stress over time, and the different demands that these stresses place on the algae. In contrast, there are 771 DEGs intersecting between MAX9vsMAX1 and MED9vsMED1, which can be attributed to the fact that these are both contrasts showing the effect of duration of treatment.

Gene ontology analysis, gene products and mineral contents

The blastx program finds comparisons between the read sequences and a protein database for all phaeophyceae. A selection of the presumed gene products that were also differentially expressed in any of the contrasts examined are visualised in the heatmaps. Keep in mind that these figures (Figures 16-17) only visualize how each gene product varies in logCPM across samples. The counts are normalized to account for the fact that the various gene products might have very different logCPM.

Iodine and vIPOs

According to the QuickGo web-tool (Bateman et al., 2021) there are four annotations on the vIPO gene product, all coming from *Laminaria digitata* (UniProtKB:Q4LDE6). Two of them are connected by the keyword “enables” to the GO term “peroxidase activity” (GO:0004601), while the other two are connected by the keyword “involved in” to the GO term “cellular oxidant detoxification” (GO:0098869. “Cellular oxidant detoxification” was among the most significant GO terms among DEGs in MAX9vsC9, and a manual search revealed that also "peroxidase activity" was included in the DEGs for MAX9vsC9 and that "cellular oxidant detoxification" was present in both MAX1vsC1 and MAX3vsC3. The fact that "peroxidase activity" was not among the top 10 for any contrast does not mean the term isn't significantly differentially expressed. Terms with -log10(p) values higher than 1.3 are significantly differentially expressed using adjusted (FDR) p-values of 0.05. In other words, genes related to peroxidase activity were up or down regulated by more than 58% compared to controls in the abovementioned contrasts. The GO terms "cellular oxidant detoxification" and "peroxidase activity" are however not exclusive to vIPOs and are connected to proteins such as glutathione reductase, catalase, APX protein, and 2-cys peroxiredoxin among others.

From the heatmap of selected gene products the relative regulation of both vIPOs and vBPOs can be seen. vIPO 1 appears to be upregulated in MAX1 samples compared control, but also compared to MAX3 and MAX9 to a lesser degree. A possible explanation why vIPO in MAX1 was upregulated compared to MAX3 and MAX9 is that other antioxidants were more active in ROS scavenging as time went by.

The iodine content measured in the mineral content analysis shows a slight uptick on day 1 for all light intensities. If vIPO is indeed upregulated in MAX1 and spending stored iodide to reduce a surplus of hydrogen peroxide, we should expect iodine content to decrease in MAX1 samples as iodide is released into the surrounding water (Küpper & Carrano, 2019). Thus, the uptick in iodine content on day 1 is slightly surprising, even if iodine was not significantly correlated with either time or light. Likely, the iodide efflux event happens on timescale of minutes or hours, and the one snapshot we have from day 1 is unlikely correspond to the peak of this response. One aspect that could be interesting to explore in future experiments is the efflux of iodine in response to stress on shorter timescale, and to monitor the iodide levels in the water.

One of the aims of this study was to investigate how light stress affected iodine content and genes related to iodine metabolism, with the broader objective of investigating whether lowering iodine content would inflict decreases stress tolerance. Ultimately, since there were no significant changes to iodine levels, it becomes difficult to draw any conclusions regarding this objective.

Bromine and vBPOs

Among the DE reads there were also reads annotated by blastx to vBPO gene products. According to the QuickGo web-tool (Binns et al., 2009), these gene products were annotated from several taxa, including two brown algae species: *S. japonica* (UniProtKB:A0A411NJS5) and *Ectocarpus siliculosus* (UniProtKB:D8LTP5). These presumed vBPOs are connected to the same GO terms as the vIPO gene product, that is it “enables” “peroxidase activity” and is “involved in” “cellular oxidant activity”.

vBPO had a reversed pattern from vIPO among the treatments, being upregulated mostly in MAX9 samples (and downregulated in C9), indicating that vBPOs are important for long-term photoprotection. However, this could also indicate that vBPOs were not important in the stress response to light and that other factors (such as time) were more prevalent at day 9.

Just as iodine, bromine content was not significantly affected by either time, light or interaction.

Heavy metals and the glutathione-ascorbate cycle

Although other antioxidants and cellular detoxification systems (catalase? Peroxidoxin?) could also have been discussed in relation to heavy metals, the glutathione peptide was chosen for this discussion because various presumed glutathione gene products upregulated and because it is believed to be important for protection against heavy metals in algae. Obviously, there was no controlled metal exposure in this experiment, so the discussion on this topic is dependent on the levels of arsenic, cadmium, and lead measured in the mineral content analysis. Also, stress response processes are typically involved in more than one type of stress.

Looking at the mineral content analysis for the light stressed individuals, both arsenic and lead was significantly affected by both time and light. From the scatterplots its apparent that both arsenic and lead levels in samples treated with 250 µmol light was lower on day 9. Although cadmium was not significantly affected by light, the same trend can be seen from the scatterplot of this element. This is an interesting find, as it does not appear to be the case for any of the other elements (non-heavy metals). It is unclear why heavy metal levels would decrease after 9 days of exposure to light stress.

Both glutathione reductase and glutathione S-transferase appear to be upregulated in treatment groups, compared to the controls, although among the different glutathione S-transferases there was no clear bias towards either MAX1, MAX3, or MAX9. The upregulation of glutathione reductase implies that there was increased oxidation in the tissues, which is as expected in stressed tissue. Since reduced glutathione is present in most of the cell, glutathione reductase activity might be an indication of the general oxidative level in the sample tissues. This assumption is based on the fact that the most prevalent form of glutathione is the reduced form but it is oxidised in highly oxidative states (Nowicka, 2022). Following this reasoning, it appears that the general level of ROS is high for the MAX1 samples, but then have come under control in MAX3, and MAX9 samples. For all time control samples, glutathione reductase appears to be at a baseline expression level.

Glutathione S-transferase, which is the enzyme associated with the linking of glutathione to heavy metals and other xenobiotics, was upregulated in treatment groups. The fact that contents of arsenic, cadmium, and lead decreased over time while exposed to intense light is agreeing with upregulation of glutathione S-transferase. In theory, this suggests that a certain level of light stress is helpful if someone endeavours only to reduce heavy metal levels in *S. latissima*. However, this is presumably not a sustainable strategy as long-term stress is likely to harm the algae in other ways.

Contents of vanadium, phosphorus, and cobalt

Vanadium and phosphorus were both negatively correlated with both time and light. While phosphorus is a macromineral central to many processes across all life, it is not especially interesting in the context of light stress. Vanadium is the namesake element in vanadium-dependent haloperoxidases, although the actual role of vanadium in vHPOs remain quite elusive in the literature. To my knowledge there is no evidence that vanadium content acts as a limiting factor in the production of vHPOs. Cobalt is the only element that was positively correlated with time. There was no significant effect of light on cobalt levels, which suggests that cobalt is not involved in light stress processes. It seems rather that the beaker residency or other factors were affecting cobalt levels in the samples.

The xanthophyll cycle, heat shock proteins, and other observations

Several GO terms and annotated gene products typically associated with stress responses were observed including the xanhophyll cycle, heat shock proteins (HSPs), and base-excision repair. It becomes clear that scavenging of ROS by vHPOs is not the only protection brown macroalgae has against abiotic stresses such as high light.

Among differentially expressed genes in samples treated with 250 µmol light for 1 day (MAX1) compared to control day 1 (C1), the most significant GO terms were “xanthophyll cycle” and “violaxanthin de-epoxidase activity” (Figure 15). The xanthophyll cycle is found throughout higher plants, as well as in green and brown algae, and is crucial for photoprotection in these groups (Goss & Latowski, 2020). The main mechanism of the cycle is characterized by violaxanthin being converted by reversible de-epoxylation to zeaxanthin, which has photoprotective abilities (Havaux & Niyogi, 1999). The composition of xanthophyll compounds (violaxanthin, zeaxanthin, lutein, and neoxanthin) is highly conserved which underscores their importance in algae and higher plants (Pogson et al., 1998). The xanthophyll cycle delivers photoprotection within minutes, but also results in extended photoprotection that can last for days, weeks, or months (Goss & Latowski, 2020). The fact that genes related to violaxanthin de-epoxidase activity were regulated in the MAX1 samples compared to the C1 samples is hardly surprising as they were exposed to intense light. However, both “xanthophyll cycle” and “violaxanthin de-epoxidase activity” scored lower in the MAX3vsC3 contrast and is not seen among the top terms in MAX9vsC9. It would seem from this result that violaxanthin de-epoxidation is high during day 1 of stress but decreases sometime between day 1 and 9. This is consistent with previous research showing that the xanthophyll cycle is fast-acting mechanism that also provides long-lasting effects (Goss & Latowski, 2020). Presumably violaxanthin de-epoxidation produces enzymes that have slow turnover (not quickly degraded). It could also suggest that other protection mechanisms are more prevalent at prolonged stress exposure.

Among the top 10 gene products found by blasting DEGs to the diamond phaeophyceae database, a presumed heat shock protein (HSP) was upregulated MAX1, MAX3, and MAX9 compared to all controls. The response was strongest on day 1 and appeared to decrease steadily until day 9. HSPs are classic stress response proteins and in a previous study HSPs were also found to be regulated in *S. latissima* after exposure to intense light at 17 °C (Heinrich, Valentin, et al., 2012).

Base-excision repair also scored high, indicating that DNA damages were occurring at an increased rate. Among the DEGs in samples exposed to 250 µmol light for 3 days compared to the day 3 control, the GO term "O-acyltransferase activity" scored very high (-log10(p) = 13.8). Generally, O-acyltransferases catalyses the transfer of acyl groups (i.e., the OH group originating from an acid molecule) to oxygen-containing acceptor molecules. However, O-acyltransferases is a huge group containing many subgroups, and the GO term "O-acyltransferase activity" has 435,928 annotations in the QuickGo database at the time of writing (Binns et al., 2009). This makes it hard to draw any specific conclusions from this finding. A study on cultivated *Saccharina latissima* found two genes encoding a probable O-acyltransferase to be downregulated after a 2-week exposure to darkness (H. Li et al., 2020).

The appearance EsV-1 is surprising. According to the QuickGo web-tool (Binns et al., 2009), this protein is marked as "located in" "integral component of membrane" (GO:0016021) of the fully sequenced brown algae *Ectocarpus siliculosus* (UniProtKB:D8LTP5). This trans-membrane protein is connected to the *Ectocarpus siliculosus* virus (EsV-1) which infects *E. siliculosus* population globally (Delaroque et al., 2000). However, the connection between the protein and the virus is not clear from the literature. This find could indicate that the EsV-1 virus also infects *S. latissima*, but I could not find any evidence from previous research to support this. The appearance of this GO term could suggest that the samples were exposed to (unplanned) biotic stress.

The GO term "peroxisome" is among the top terms in MAX1vsC1, which is interesting in the context of vHPOs. Peroxisomes are the organelles usually associated with the oxidation processes that produce hydrogen peroxide, which is subsequently scavenged by either catalase or enzymes like vHPOs in algae. Several types of substrates are broken down by such oxidation reactions in the peroxisome, including uric acid and fatty acids (Cooper & Hausman, 2007). To the best of the authors knowledge, there is no consensus on the subcellular distribution of vHPOs in algae, and it is not inconceivable that peroxisomes have high concentrations, although this is just a speculation. Peroxisome genes being regulated could also mean for one that other peroxidases like ascorbate peroxidase (APX) were active in the peroxisome.

Conclusion

There was found significant variations in mineral content between the stipe and bottom samples in S. latissima. Iodine, arsenic, phosphorus, and vanadium levels were also affected by rope section (measure of depth) in bottom samples. Individuals growing closer to the surface tended to have more biomass (fresh weight) than individuals growing deeper, but the relationship was not significant at $p = 0.05$. This research indicates that the scavenging of ROS by vIPO and vBPO is an important light stress response in *S. latissima*, but also that other processes such as the xanthophyll cycle and the glutathione-ascorbate cycle is important in this regard. Interestingly, the upregulation of vIPO in MAX1 samples did not correspond to an efflux of iodine.

The relationship between halogen content, vHPOs and stress in *S. latissima* should be explored further in future studies. Maybe vIPO regulation and the efflux of iodine in response to stress should be studied on a shorter timescale. It should be considered in future light stress experiments on S. latissima that 100 µmol light might not be enough to cause considerate DE.

Regarding the data in this study, several aspects can be explored further. New de novo assemblies can be made as discussed and compared to the current. Gene enrichment analyses and KEGG pathway analyses can be carried out which would aid in understanding the transcriptomic processes observed here. In addition, certain selected parameters like count filter and log fold change threshold can be experimented with to increase the scope of the study and to look other aspects of the transcriptome.

References

Agilent Technologies Inc. (2015). Agilent D1000 ScreenTape System Quick Guide. *Agilent Technologies Inc.*, *3*.

Ahmad, P., Jaleel, C. A., Salem, M. A., Nabi, G., & Sharma, S. (2010). Roles of enzymatic and nonenzymatic antioxidants in plants during abiotic stress. In *Critical Reviews in Biotechnology* (Vol. 30, Issue 3). https://doi.org/10.3109/07388550903524243

Almeida, M., Filipe, S., Humanes, M., Maia, M. F., Melo, R., Severino, N., da Silva, J. A. L., Fraústo da Silva, J. J. R., & Wever, R. (2001). Vanadium haloperoxidases from brown algae of the Laminariaceae family. *Phytochemistry*. https://doi.org/10.1016/S0031-9422(01)00094-2

Almela, C., Jesús Clemente, M., Vélez, D., & Montoro, R. (2006). Total arsenic, inorganic arsenic, lead and cadmium contents in edible seaweed sold in Spain. *Food and Chemical Toxicology*, *44*(11). https://doi.org/10.1016/j.fct.2006.06.011

Amat, M. A., & Srivastava, L. M. (1985). TRANSLOCATION OF IODINE IN LAMINARIA SACCHARINA(PHAEOPHYTA). *Journal of Phycology*. https://doi.org/10.1111/j.0022-3646.1985.00330.x

Anders, S., & Huber, W. (2010). Differential expression analysis for sequence count data. *Genome Biology*, *11*(10). https://doi.org/10.1186/gb-2010-11-10-r106

Andrews, S. (2010). FastQC - A quality control tool for high throughput sequence data. http://www.bioinformatics.babraham.ac.uk/projects/fastqc/. *Babraham Bioinformatics*.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., & Sherlock, G. (2000). Gene ontology: Tool for the unification of biology. In *Nature Genetics* (Vol. 25, Issue 1). https://doi.org/10.1038/75556

Bateman, A., Martin, M. J., Orchard, S., Magrane, M., Agivetova, R., Ahmad, S., Alpi, E., Bowler-Barnett, E. H., Britto, R., Bursteinas, B., Bye-A-Jee, H., Coetzee, R., Cukura, A., da Silva, A., Denny, P., Dogan, T., Ebenezer, T. G., Fan, J., Castro, L. G., … Teodoro, D. (2021). UniProt: the universal protein knowledgebase in 2021. *Nucleic Acids Research*, *49*(D1). https://doi.org/10.1093/nar/gkaa1100

Bekkby, T., & Moy, F. E. (2011). Developing spatial models of sugar kelp (Saccharina latissima) potential distribution under natural conditions and areas of its disappearance in Skagerrak. *Estuarine, Coastal and Shelf Science*. https://doi.org/10.1016/j.ecss.2011.10.029

Berglin, M., Delage, L., Potin, P., Vilter, H., & Elwing, H. (2004). Enzymatic cross-linking of a phenolic polymer extracted from the marine alga Fucus serratus. *Biomacromolecules*, *5*(6). https://doi.org/10.1021/bm0496864

Binns, D., Dimmer, E., Huntley, R., Barrell, D., O'Donovan, C., & Apweiler, R. (2009). QuickGO: A web-based tool for Gene Ontology searching. *Bioinformatics*, *25*(22). https://doi.org/10.1093/bioinformatics/btp536

Bolger, A. M., Lohse, M., & Usadel, B. (2014). Trimmomatic: A flexible trimmer for Illumina sequence data. *Bioinformatics*, *30*(15). https://doi.org/10.1093/bioinformatics/btu170

Bougeard, S., & Dray, S. (2018). Supervised multiblock analysis in R with the ade4 package. *Journal of Statistical Software*, *86*. https://doi.org/10.18637/jss.v086.i01

Buchfink, B., Reuter, K., & Drost, H. G. (2021). Sensitive protein alignments at tree-of-life scale using DIAMOND. *Nature Methods*, *18*(4). https://doi.org/10.1038/s41592-021-01101-x

Buschmann, A. H., Camus, C., Infante, J., Neori, A., Israel, Á., Hernández-González, M. C., Pereda, S. v., Gomez-Pinchetti, J. L., Golberg, A., Tadmor-Shalev, N., & Critchley, A. T. (2017). Seaweed production: overview of the global state of exploitation, farming and emerging research activity. *European Journal of Phycology*, *52*(4). https://doi.org/10.1080/09670262.2017.1365175

Carbon, S., Douglass, E., Good, B. M., Unni, D. R., Harris, N. L., Mungall, C. J., Basu, S., Chisholm, R. L., Dodson, R. J., Hartline, E., Fey, P., Thomas, P. D., Albou, L. P., Ebert, D., Kesling, M. J., Mi, H., Muruganujan, A., Huang, X., Mushayahama, T., … Elser, J. (2021). The Gene Ontology resource: Enriching a GOld mine. *Nucleic Acids Research*, *49*(D1). https://doi.org/10.1093/nar/gkaa1113

Caverzan, A., Passaia, G., Rosa, S. B., Ribeiro, C. W., Lazzarotto, F., & Margis-Pinheiro, M. (2012). Plant responses to stresses: Role of ascorbate peroxidase in the antioxidant protection. In *Genetics and Molecular Biology* (Vol. 35, Issue 4 SUPPL.). https://doi.org/10.1590/S1415-47572012000600016

Charif, D., Lobry, J. R., Necsulea, A., Palmeira, L., Perriere, G., & Penel, M. S. (2015). Package seqinr. *R Package*.

Chen, Y., Lun, A. T. L., & Smyth, G. K. (2016). From reads to genes to pathways: Differential expression analysis of RNA-Seq experiments using Rsubread and the edgeR quasi-likelihood pipeline. *F1000Research*, *5*. https://doi.org/10.12688/F1000RESEARCH.8987.2

Chessel, D., Dufour, A. B., & Thioulouse, J. (2004). The ade4 Package --- {I}: One-table Methods. *R News*, *4*(1).

Cock, J. M., Sterck, L., Rouzé, P., Scornet, D., Allen, A. E., Amoutzias, G., Anthouard, V., Artiguenave, F., Aury, J. M., Badger, J. H., Beszteri, B., Billiau, K., Bonnet, E., Bothwell, J. H., Bowler, C., Boyen, C., Brownlee, C., Carrano, C. J., Charrier, B., … Wincker, P. (2010). The Ectocarpus genome and the independent evolution of multicellularity in brown algae. *Nature*, *465*(7298). https://doi.org/10.1038/nature09016

Colin, C., Leblanc, C., Michel, G., Wagner, E., Leize-Wagner, E., van Dorsselaer, A., & Potin, P. (2005). Vanadium-dependent iodoperoxidases in Laminaria digitata, a novel biochemical function diverging from brown algal bromoperoxidases. *Journal of Biological Inorganic Chemistry*, *10*(2). https://doi.org/10.1007/s00775-005-0626-8

Colin, C., Leblanc, C., Wagner, E., Delage, L., Leize-Wagner, E., van Dorsselaer, A., Kloareg, B., & Potin, P. (2003). The brown algal kelp Laminaria digitata features distinct bromoperoxidase and iodoperoxidase activities. *Journal of Biological Chemistry*, *278*(26). https://doi.org/10.1074/jbc.M300247200

Cooper, G. M., & Hausman, R. E. (2007). The Cell: A Molecular Approach 2nd Edition. In *Sinauer Associates*.

Crans, D. C., Smee, J. J., Gaidamauskas, E., & Yang, L. (2004). The Chemistry and Biochemistry of Vanadium and the Biological Activities Exerted by Vanadium Compounds. In *Chemical Reviews* (Vol. 104, Issue 2). https://doi.org/10.1021/cr020607t

Davison, I. R., Jordan, T. L., Fegley, J. C., & Grobe, C. W. (2007). Response of Laminaria saccharina (Phaeophyta) growth and photosynthesis to simultaneous ultraviolet radiation and nitrogen limitation. *Journal of Phycology*, *43*(4). https://doi.org/10.1111/j.1529-8817.2007.00360.x

Delaroque, N., Wolf, S., Muller, D. G., & Knippers, R. (2000). The brown algal virus EsV-1 particle contains a putative hybrid histidine kinase. *Virology*, *273*(2). https://doi.org/10.1006/viro.2000.0383

Díaz, O., Tapia, Y., Muñoz, O., Montoro, R., Velez, D., & Almela, C. (2012). Total and inorganic arsenic concentrations in different species of economically important algae harvested from coastal zones of Chile. *Food and Chemical Toxicology*, *50*(3–4). https://doi.org/10.1016/j.fct.2011.11.024

Dray, S., & Dufour, A. B. (2007). The ade4 package: Implementing the duality diagram for ecologists. *Journal of Statistical Software*, *22*(4). https://doi.org/10.18637/jss.v022.i04

Dray, S., Dufour, A.-B., & Chessel, D. (2007). The ade4 package - II: Two-table and K-table methods. *R News*, *7*(2).

European Commission. (2015). COMMISSION REGULATION (EU) 2015/1006 of 25 June 2015 amending Regulation (EC) No 1881/2006 as regards maximum levels of inorganic arsenic in foodstuff. *Official Journal of the European Union*.

Ewels, P., Magnusson, M., Lundin, S., & Käller, M. (2016). MultiQC: Summarize analysis results for multiple tools and samples in a single report. *Bioinformatics*, *32*(19). https://doi.org/10.1093/bioinformatics/btw354

FAO, & WHO. (2001). Human Vitamin and Mineral Requirements. *Human Vitamin and Mineral Requirements*.

Getachew, P., Kang, J. Y., Choi, J. S., & Hong, Y. K. (2015). Does bryozoan colonization alter the biochemical composition of Saccharina japonica affecting food safety and quality? *Botanica Marina*, *58*(4). https://doi.org/10.1515/bot-2015-0007

Ginestet, C. (2011). ggplot2: Elegant Graphics for Data Analysis. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, *174*(1). https://doi.org/10.1111/j.1467-985x.2010.00676_9.x

Goecke, F., Klemetsdal, G., & Ergon, Å. (2020). Cultivar Development of Kelps for Commercial Cultivation—Past Lessons and Future Prospects. In *Frontiers in Marine Science*. https://doi.org/10.3389/fmars.2020.00110

Goss, R., & Latowski, D. (2020). Lipid Dependence of Xanthophyll Cycling in Higher Plants and Algae. In *Frontiers in Plant Science* (Vol. 11). https://doi.org/10.3389/fpls.2020.00455

Götz, S., García-Gómez, J. M., Terol, J., Williams, T. D., Nagaraj, S. H., Nueda, M. J., Robles, M., Talón, M., Dopazo, J., & Conesa, A. (2008). High-throughput functional annotation and data mining with the Blast2GO suite. *Nucleic Acids Research*, *36*(10). https://doi.org/10.1093/nar/gkn176

Grabherr, M. G., Haas, B. J., Yassour, M., Levin, J. Z., Thompson, D. A., Amit, I., Adiconis, X., Fan, L., Raychowdhury, R., Zeng, Q., Chen, Z., Mauceli, E., Hacohen, N., Gnirke, A., Rhind, N., di Palma, F., Birren, B. W., Nusbaum, C., Lindblad-Toh, K., … Regev, A. (2011). Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nature Biotechnology*, *29*(7). https://doi.org/10.1038/nbt.1883

Haas, B. J., Papanicolaou, A., Yassour, M., Grabherr, M., Blood, P. D., Bowden, J., Couger, M. B., Eccles, D., Li, B., Lieber, M., Macmanes, M. D., Ott, M., Orvis, J., Pochet, N., Strozzi, F., Weeks, N., Westerman, R., William, T., Dewey, C. N., … Regev, A. (2013). De novo transcript sequence reconstruction from RNA-seq using the Trinity platform for reference generation and analysis. *Nature Protocols*, *8*(8). https://doi.org/10.1038/nprot.2013.084

Havaux, M., & Niyogi, K. K. (1999). The violaxanthin cycle protects plants from photooxidative damage by more than one mechanism. *Proceedings of the National Academy of Sciences of the United States of America*, *96*(15). https://doi.org/10.1073/pnas.96.15.8762

Heinrich, S., Frickenhaus, S., Glöckner, G., & Valentin, K. (2012). A comprehensive cDNA library of light- and temperature-stressed Saccharina latissima (Phaeophyceae). *European Journal of Phycology*, *47*(2). https://doi.org/10.1080/09670262.2012.660639

Heinrich, S., Valentin, K., Frickenhaus, S., John, U., & Wiencke, C. (2012). Transcriptomic Analysis of Acclimation to Temperature and Light Stress in Saccharina latissima (Phaeophyceae). *PLoS ONE*, *7*(8). https://doi.org/10.1371/journal.pone.0044342

Krela-Kaźmierczak, I., Czarnywojtek, A., Skoracka, K., Rychter, A. M., Ratajczak, A. E., Szymczak-Tomczak, A., Ruchała, M., & Dobrowolska, A. (2021). Is there an ideal diet to protect against iodine deficiency? In *Nutrients* (Vol. 13, Issue 2). https://doi.org/10.3390/nu13020513

Kukurba, K. R., & Montgomery, S. B. (2015). RNA sequencing and analysis. *Cold Spring Harbor Protocols*, *2015*(11), 951–969. https://doi.org/10.1101/pdb.top084970

Küpper, F. C., Carpenter, L. J., Leblanc, C., Toyama, C., Uchida, Y., Maskrey, B. H., Robinson, J., Verhaeghe, E. F., Malin, G., Luther, G. W., Kroneck, P. M. H., Kloareg, B., Meyer-Klaucke, W., Muramatsu, Y., Megson, I. L., Potin, P., & Feiters, M. C. (2013). In vivo speciation studies and antioxidant properties of bromine in Laminaria digitata reinforce the significance of iodine accumulation for kelps. *Journal of Experimental Botany*. https://doi.org/10.1093/jxb/ert110

Küpper, F. C., Carpenter, L. J., McFiggans, G. B., Palmer, C. J., Waite, T. J., Boneberg, E. M., Woitsch, S., Weiller, M., Abela, R., Grolimund, D., Potin, P., Butler, A., Luther, G. W., Kroneck, P. M. H., Meyer-Klaucke, W., & Feiters, M. C. (2008). Iodide accumulation provides kelp with an inorganic antioxidant impacting atmospheric chemistry. *Proceedings of the National Academy of Sciences of the United States of America*. https://doi.org/10.1073/pnas.0709959105

Küpper, F. C., & Carrano, C. J. (2019). Key aspects of the iodine metabolism in brown algae: A brief critical review. In *Metallomics*. https://doi.org/10.1039/c8mt00327k

Küpper, F. C., Schweigert, N., Ar Gall, E., Legendre, J. M., Vilter, H., & Kloareg, B. (1998). Iodine uptake in Laminariales involves extracellular, haloperoxidase-mediated oxidation of iodide. *Planta*, *207*(2). https://doi.org/10.1007/s004250050469

Langmead, B., & Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. *Nature Methods*, *9*(4). https://doi.org/10.1038/nmeth.1923

Leung, A. M., & Braverman, L. E. (2014). Consequences of excess iodine. In *Nature Reviews Endocrinology*. https://doi.org/10.1038/nrendo.2013.251

Li, B., & Dewey, C. N. (2011). RSEM: Accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC Bioinformatics*, *12*. https://doi.org/10.1186/1471-2105-12-323

Li, H., Scheschonk, L., Heinrich, S., Valentin, K., Harms, L., Glöckner, G., Corre, E., & Bischof, K. (2020). Transcriptomic Responses to Darkness and the Survival Strategy of the Kelp Saccharina latissima in the Early Polar Night. *Frontiers in Marine Science*, *7*. https://doi.org/10.3389/fmars.2020.592033

Liang, X., Wang, X., Chi, S., Wu, S., Sun, J., Liu, C., Chen, S., Yu, J., & Liu, T. (2014). Analysis of Saccharina japonica transcriptome using the high-throughput DNA sequencing technique and its vanadium-dependent haloperoxidase gene. *Acta Oceanologica Sinica*. https://doi.org/10.1007/s13131-014-0438-1

Liu, T., Wang, X., Wang, G., Jia, S., Liu, G., Shan, G., Chi, S., Zhang, J., Yu, Y., Xue, T., & Yu, J. (2019). Evolution of complex thallus alga: Genome Sequencing of Saccharina japonica. *Frontiers in Genetics*, *10*(MAY). https://doi.org/10.3389/fgene.2019.00378

Love, M. I., Huber, W., & Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*, *15*(12). https://doi.org/10.1186/s13059-014-0550-8

Lydia Gieselmann. (n.d.). *Saccharina latissima thallus*. Silje Forbord.

Maharana, D., Das, P. B., Verlecar, X. N., Pise, N. M., & Gauns, M. (2015). Oxidative stress tolerance in intertidal red seaweed Hypnea musciformis (Wulfen) in relation to environmental components. *Environmental Science and Pollution Research*. https://doi.org/10.1007/s11356-015-4985-6

Manni, M., Berkeley, M. R., Seppey, M., Simão, F. A., & Zdobnov, E. M. (2021). BUSCO Update: Novel and Streamlined Workflows along with Broader and Deeper Phylogenetic Coverage for Scoring of Eukaryotic, Prokaryotic, and Viral Genomes. *Molecular Biology and Evolution*, *38*(10). https://doi.org/10.1093/molbev/msab199

McCarthy, D. J., Chen, Y., & Smyth, G. K. (2012). Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation. *Nucleic Acids Research*, *40*(10). https://doi.org/10.1093/nar/gks042

Monteiro, C. M. M. H., Li, H., Bischof, K., Bartsch, I., Valentin, K. U., Corre, E., Collén, J., Harms, L., Glöckner, G., & Heinrich, S. (2019). Is geographical variation driving the transcriptomic responses to multiple stressors in the kelp Saccharina latissima? *BMC Plant Biology*, *19*(1). https://doi.org/10.1186/s12870-019-2124-0

Nagalakshmi, U., Wang, Z., Waern, K., Shou, C., Raha, D., Gerstein, M., & Snyder, M. (2008). The transcriptional landscape of the yeast genome defined by RNA sequencing. *Science*, *320*(5881). https://doi.org/10.1126/science.1158441

Nanodrop. (1975). 260/280 and 260/230 Ratios. *Ratio*, *17*.

Nielsen, C. W., Holdt, S. L., Sloth, J. J., Marinho, G. S., Sæther, M., Funderud, J., & Rustad, T. (2020). Reducing the high iodine content of Saccharina latissima and improving the profile of other valuable compounds by water blanching. *Foods*. https://doi.org/10.3390/foods9050569

No, E. C. (1881). Commission regulation (EC) No. 1881/2006 of 19 December 2006. Setting maximum levels for certain contaminants in foodstuffs (Text with EEA relevance). *Official Journal of European Commission*, *364*(20.12).

Nowicka, B. (2022). Heavy metal–induced stress in eukaryotic algae—mechanisms of heavy metal toxicity and tolerance with particular emphasis on oxidative stress in exposed cells and the role of antioxidant response. In *Environmental Science and Pollution Research* (Vol. 29, Issue 12). https://doi.org/10.1007/s11356-021-18419-w

Pawlik-Skowrońska, B., Pirszel, J., & Brown, M. T. (2007). Concentrations of phytochelatins and glutathione found in natural assemblages of seaweeds depend on species and metal concentrations of the habitat. *Aquatic Toxicology*, *83*(3). https://doi.org/10.1016/j.aquatox.2007.04.003

Pearson, W. R., & Lipman, D. J. (1988). Improved tools for biological sequence comparison. *Proceedings of the National Academy of Sciences of the United States of America*, *85*(8). https://doi.org/10.1073/pnas.85.8.2444

Pétursdóttir, Á. H., Blagden, J., Gunnarsson, K., Raab, A., Stengel, D. B., Feldmann, J., & Gunnlaugsdóttir, H. (2019). Arsenolipids are not uniformly distributed within two brown macroalgal species Saccharina latissima and Alaria esculenta. *Analytical and Bioanalytical Chemistry*. https://doi.org/10.1007/s00216-019-01907-x

Petursdottir, A. H., Sloth, J. J., & Feldmann, J. (2015). Introduction of regulations for arsenic in feed and food with emphasis on inorganic arsenic, and implications for analytical chemistry. *Analytical and Bioanalytical Chemistry*. https://doi.org/10.1007/s00216-015-9019-1

Pogson, B. J., Niyogi, K. K., Björkman, O., & DellaPenna, D. (1998). Altered xanthophyll compositions adversely affect chlorophyll accumulation and nonphotochemical quenching in Arabidopsis mutants. *Proceedings of the National Academy of Sciences of the United States of America*, *95*(22). https://doi.org/10.1073/pnas.95.22.13324

Pospíšil, P. (2016). Production of reactive oxygen species by photosystem II as a response to light and temperature stress. *Frontiers in Plant Science*. https://doi.org/10.3389/fpls.2016.01950

QIAGEN. (2012). RNeasy Mini Handbook. *Sample & Assay Technologies*, *June*.

R Team, C. (2021). R: A Language and Environment for Statistical Computing. In *R Foundation for Statistical Computing* (4.1.2 (2021-11-01)).

Ritchie, M. E., Phipson, B., Wu, D., Hu, Y., Law, C. W., Shi, W., & Smyth, G. K. (2015). Limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Research*, *43*(7). https://doi.org/10.1093/nar/gkv007

Robinson, M. D., McCarthy, D. J., & Smyth, G. K. (2009). edgeR: A Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, *26*(1). https://doi.org/10.1093/bioinformatics/btp616

Robinson, M. D., & Oshlack, A. (2010). A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biology*, *11*(3). https://doi.org/10.1186/gb-2010-11-3-r25

Roleda, M. Y., Hanelt, D., & Wiencke, C. (2006). Growth and DNA damage in young Laminaria sporophytes exposed to ultraviolet radiation: Implication for depth zonation of kelps on Helgoland (North Sea). *Marine Biology*, *148*(6). https://doi.org/10.1007/s00227-005-0169-0

Roleda, M. Y., Marfaing, H., Desnica, N., Jónsdóttir, R., Skjermo, J., Rebours, C., & Nitschke, U. (2019). Variations in polyphenol and heavy metal contents of wild-harvested and cultivated seaweed bulk biomass: Health risk assessment and implication for food applications. *Food Control*. https://doi.org/10.1016/j.foodcont.2018.07.031

Roleda, M. Y., Skjermo, J., Marfaing, H., Jónsdóttir, R., Rebours, C., Gietl, A., Stengel, D. B., & Nitschke, U. (2018). Iodine content in bulk biomass of wild-harvested and cultivated edible seaweeds: Inherent variations determine species-specific daily allowable consumption. *Food Chemistry*. https://doi.org/10.1016/j.foodchem.2018.02.024

Roleda, M. Y., Wiencke, C., Hanelt, D., & Bischof, K. (2007). Sensitivity of the Early Life Stages of Macroalgae from the Northern Hemisphere to Ultraviolet Radiation†. *Photochemistry and Photobiology*, *83*(4). https://doi.org/10.1562/2006-08-17-ir-1005

Sandy, M., Carter-Franklin, J. N., Martin, J. D., & Butler, A. (2011). Vanadium bromoperoxidase from Delisea pulchra: Enzyme-catalyzed formation of bromofuranone and attendant disruption of quorum sensing. *Chemical Communications*, *47*(44). https://doi.org/10.1039/c1cc15605e

Shanmugam, R. (2020). Multivariate analysis of ecological data with ade4. *Journal of Statistical Computation and Simulation*, *90*(6). https://doi.org/10.1080/00949655.2019.1628898

Sharma, S., Neves, L., Funderud, J., Mydland, L. T., Øverland, M., & Horn, S. J. (2018). Seasonal and depth variations in the chemical composition of cultivated Saccharina latissima. *Algal Research*. https://doi.org/10.1016/j.algal.2018.03.012

Spurkland, T., & Iken, K. (2011). Salinity and irradiance effects on growth and maximum photosynthetic quantum yield in subarctic Saccharina latissima (Laminariales, Laminariaceae). *Botanica Marina*, *54*(4). https://doi.org/10.1515/BOT.2011.042

Stévant, P., Rebours, C., & Chapman, A. (2017). Seaweed aquaculture in Norway: recent industrial developments and future perspectives. In *Aquaculture International* (Vol. 25, Issue 4). https://doi.org/10.1007/s10499-017-0120-7

Teagle, H., Hawkins, S. J., Moore, P. J., & Smale, D. A. (2017). The role of kelp species as biogenic habitat formers in coastal marine ecosystems. In *Journal of Experimental Marine Biology and Ecology* (Vol. 492). https://doi.org/10.1016/j.jembe.2017.01.017

Thermo Fisher Scientific Inc. (2008). *NanoDrop 1000 Spectrophotometer, User's manual*. http://www.nanodrop.com/library/nd-1000-v3.7-users-manual-8.5x11.pdf

Verhaeghe, E. F., Fraysse, A., Guerquin-Kern, J. L., Wu, T. di, Devès, G., Mioskowski, C., Leblanc, C., Ortega, R., Ambroise, Y., & Potin, P. (2008). Microchemical imaging of iodine distribution in the brown alga Laminaria digitata suggests a new mechanism for its accumulation. *Journal of Biological Inorganic Chemistry*, *13*(2). https://doi.org/10.1007/s00775-007-0319-6

Weber, A. P. M. (2015). Discovering new biology through sequencing of RNA. *Plant Physiology*, *169*(3). https://doi.org/10.1104/pp.15.01081

Wickham, H. (2019). Package stringr: Simple, Consistent Wrappers for Common String Operations. In *Cran*.

Wickham, H. (2022). *Waldo: Find Differences Between R Objects.* (0.4.0.). R package. https://CRAN.R-project.org/package=waldo

Wickham, H., François, R., Henry, L., & Müller, K. (2019). dplyr: A Grammar of Data Manipulation. R package version. In *Media*.

Wickham, H., & Henry, L. (2019). tidyr: Tidy Messy Data. In *R package version 1.0.0*.

Wiencke, C., Roleda, M. Y., Gruber, A., Clayton, M. N., & Bischof, K. (2006). Susceptibility of zoospores to UV radiation determines upper depth distribution limit of Arctic kelps: Evidence through field experiments. *Journal of Ecology*, *94*(2). https://doi.org/10.1111/j.1365-2745.2006.01102.x

Wood, D. E., Lu, J., & Langmead, B. (2019). Improved metagenomic analysis with Kraken 2. *Genome Biology*, *20*(1). https://doi.org/10.1186/s13059-019-1891-0

Ye, N., Zhang, X., Miao, M., Fan, X., Zheng, Y., Xu, D., Wang, J., Zhou, L., Wang, D., Gao, Y., Wang, Y., Shi, W., Ji, P., Li, D., Guan, Z., Shao, C., Zhuang, Z., Gao, Z., Qi, J., & Zhao, F. (2015). Saccharina genomes provide novel insight into kelp biology. *Nature Communications*, *6*. https://doi.org/10.1038/ncomms7986

Zhang, Q. S., Tang, X. X., Cong, Y. Z., Qu, S. C., Luo, S. J., & Yang, G. P. (2007). Breeding of an elite Laminaria variety 90-1 through inter-specific gametophyte crossing. *Journal of Applied Phycology*, *19*(4). https://doi.org/10.1007/s10811-006-9137-4

Appendix I

Table 1. Subsets in study 1.

Name	Number of samples	Definition	Purpose
Subset 1	20	Contains both stipe and bottom samples from 10 individuals.	Look for differences between thallus part.

Subset 2	10	Contains only bottom sample from 10 individuals.	Look for variance that is not due to part.

MultiQC

Table 2: MultiQC output, general stats.

Sample name	% dups	% GC	Million seqs
K10_1	71.10%	56%	49.4
K10_2	70.80%	56%	49.4
K11_1	83.10%	54%	51.7
K11_2	82.40%	54%	51.7
K12_1	65.80%	55%	25.6
K12_2	65.60%	55%	25.6
K13_1	68.40%	54%	31.5
K13_2	67.10%	54%	31.5
K14_1	70.50%	55%	28.1
K14_2	69.90%	56%	28.1
K15_1	77.80%	55%	54.3
K15_2	77.80%	55%	54.3
K16_1	63.70%	55%	23.9
K16_2	62.30%	55%	23.9
K17_1	65.40%	55%	24.2
K17_2	64.30%	55%	24.2
K18_1	60.50%	54%	21.1
K18_2	59.20%	54%	21.1
K19_1	70.50%	54%	27.2
K19_2	69.30%	54%	27.2
K1_1	61.90%	54%	22.8
K1_2	61.40%	54%	22.8
K20_1	67.40%	55%	20
K20_2	66.00%	55%	20
K21_1	62.50%	55%	23.7
K21_2	63.40%	55%	23.7
K22_1	48.50%	54%	24.4
K22_2	51.10%	54%	24.4
K23_1	65.10%	54%	22.7
K23_2	63.90%	54%	22.7
K24_1	65.80%	55%	27.4
K24_2	66.10%	55%	27.4
K25_1	70.20%	53%	25.3
K25_2	69.30%	54%	25.3
K26_1	63.70%	54%	21
K26_2	62.90%	54%	21

K27_1	68.20%	54%	21.2
K27_2	67.20%	54%	21.2
K28_1	70.00%	54%	19.8
K28_2	69.20%	54%	19.8
K29_1	64.10%	54%	21.3
K29_2	64.40%	54%	21.3
K2_1	68.20%	55%	19.5
K2_2	67.30%	55%	19.5
K30_1	67.80%	54%	22.6
K30_2	68.10%	54%	22.6
K31_1	68.80%	54%	26
K31_2	69.30%	54%	26
K32_1	67.70%	54%	21.8
K32_2	67.40%	54%	21.8
K33_1	73.10%	54%	29
K33_2	73.20%	54%	29
K34_1	66.60%	55%	31
K34_2	65.20%	55%	31
K35_1	83.30%	53%	44.9
K35_2	83.10%	53%	44.9
K36_1	71.30%	54%	25.7
K36_2	71.70%	54%	25.7
K37_1	76.30%	55%	50.4
K37_2	75.80%	55%	50.4
K38_1	65.90%	55%	22.1
K38_2	64.90%	55%	22.1
K39_1	71.40%	53%	22
K39_2	71.10%	53%	22
K3_1	75.30%	55%	23.7
K3_2	74.50%	56%	23.7
K40_1	66.90%	54%	26.7
K40_2	66.50%	54%	26.7
K41_1	26.70%	53%	21.2
K41_2	38.90%	54%	21.2
K42_1	63.50%	54%	19.6
K42_2	62.40%	54%	19.6
K43_1	65.90%	54%	21.3
K43_2	65.90%	54%	21.3
K44_1	70.00%	53%	21.8
K44_2	69.10%	53%	21.8
K4_1	60.40%	55%	24.1
K4_2	57.10%	55%	24.1
K5_1	65.80%	55%	21.1
K5_2	64.70%	55%	21.1
K6_1	72.40%	55%	25.2
K6_2	70.30%	55%	25.2
K7_1	66.30%	55%	23.9
K7_2	64.50%	56%	23.9

K8_1	65.70%	54%	24.8
K8_2	64.50%	55%	24.8

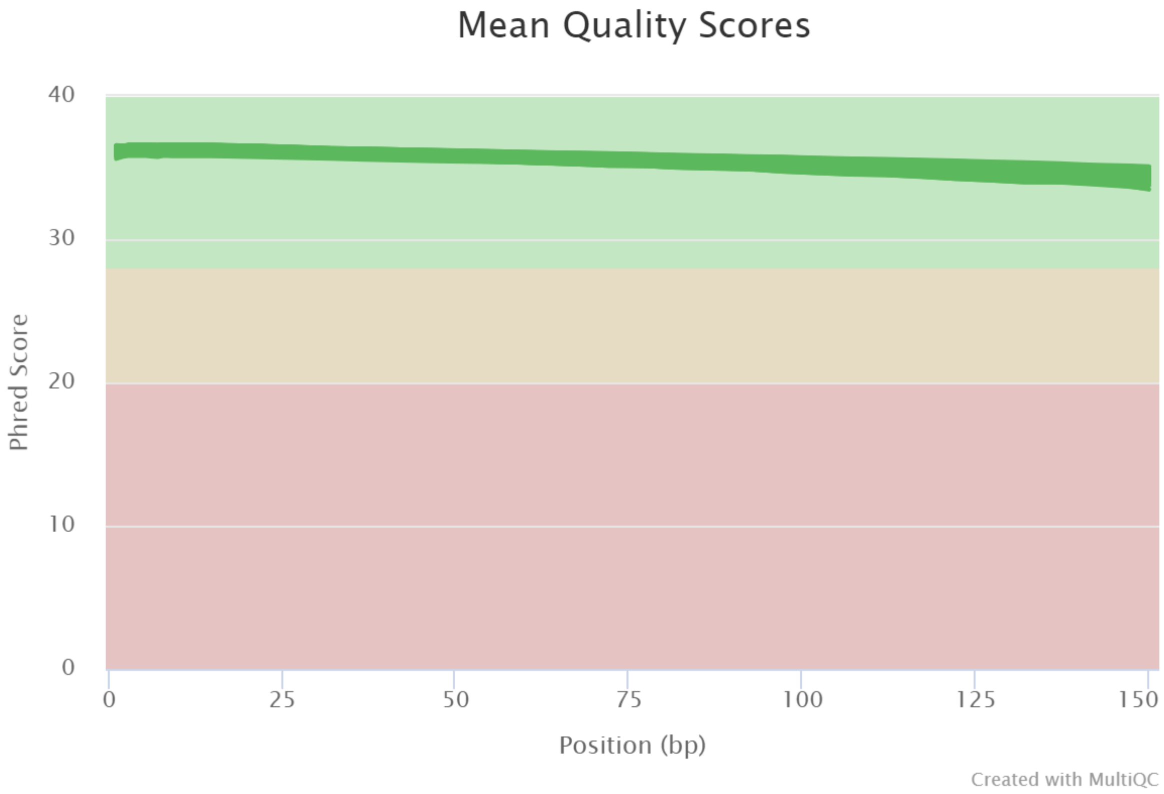


Figure 1: MultiQC output, mean quality scores.

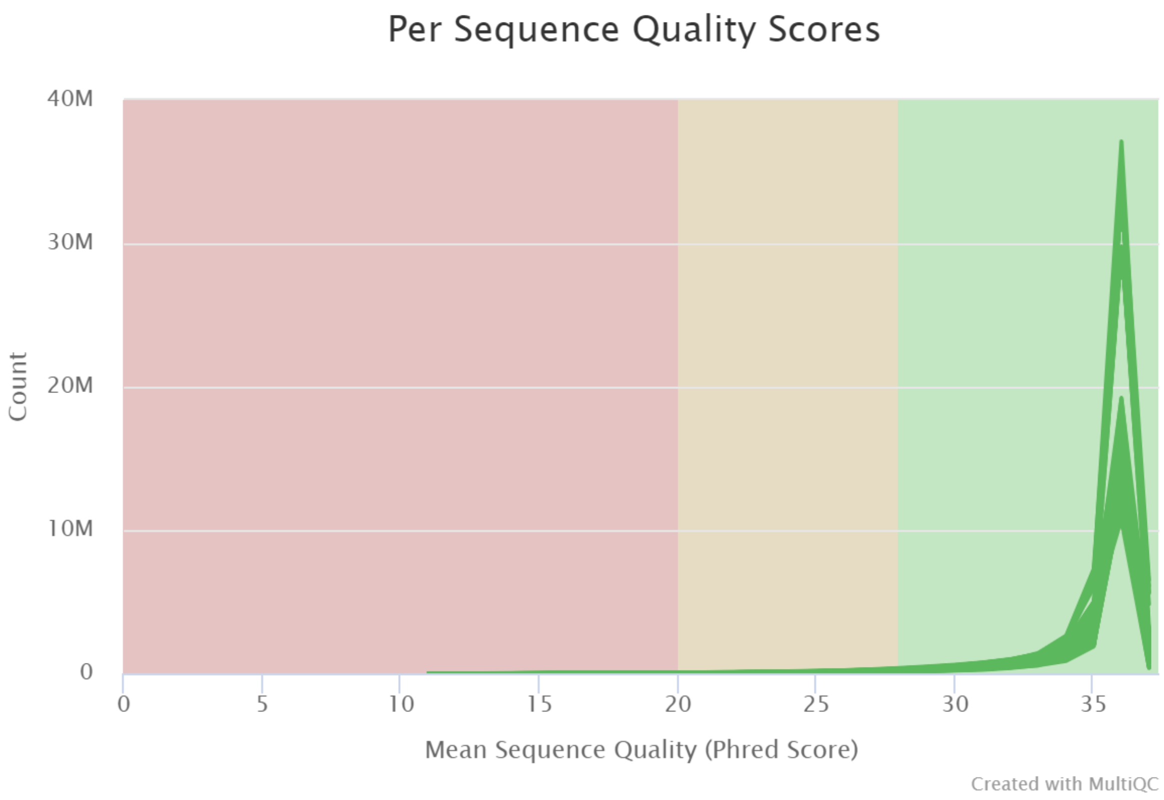


Figure 2: MultiQC output, per sequence quality scores.

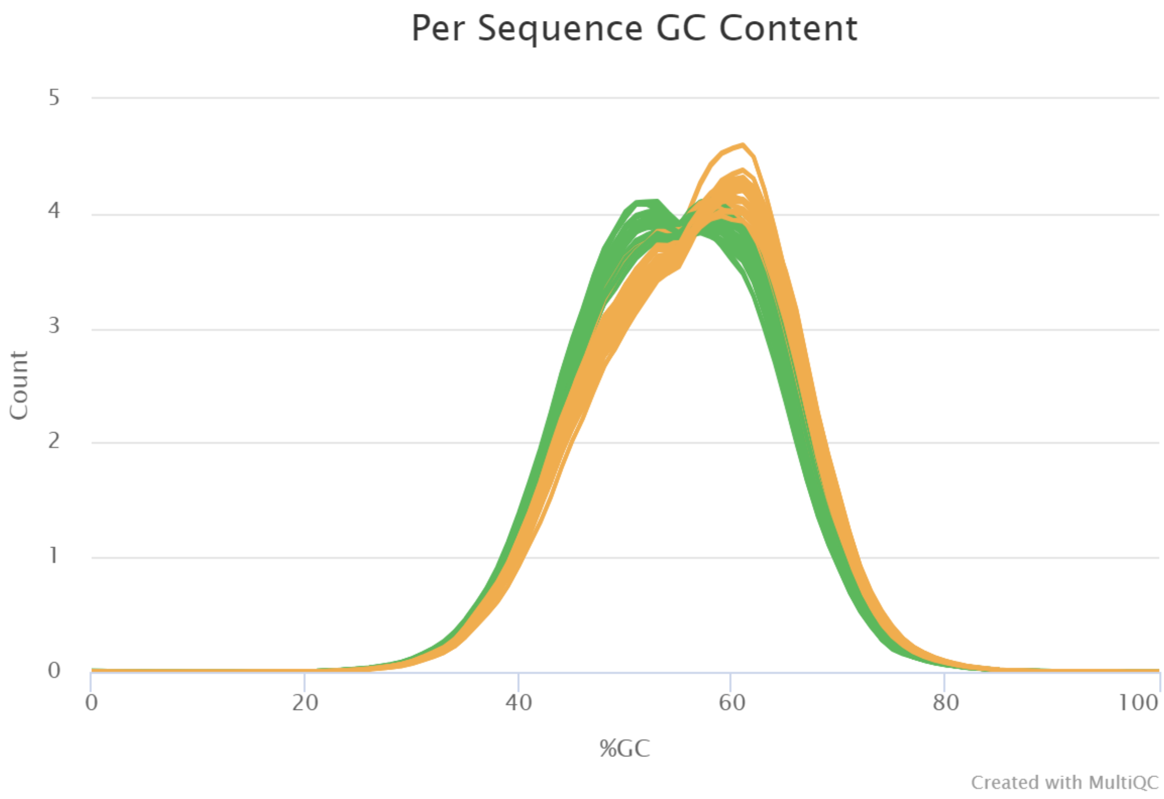


Figure 3: Per sequence CG content.

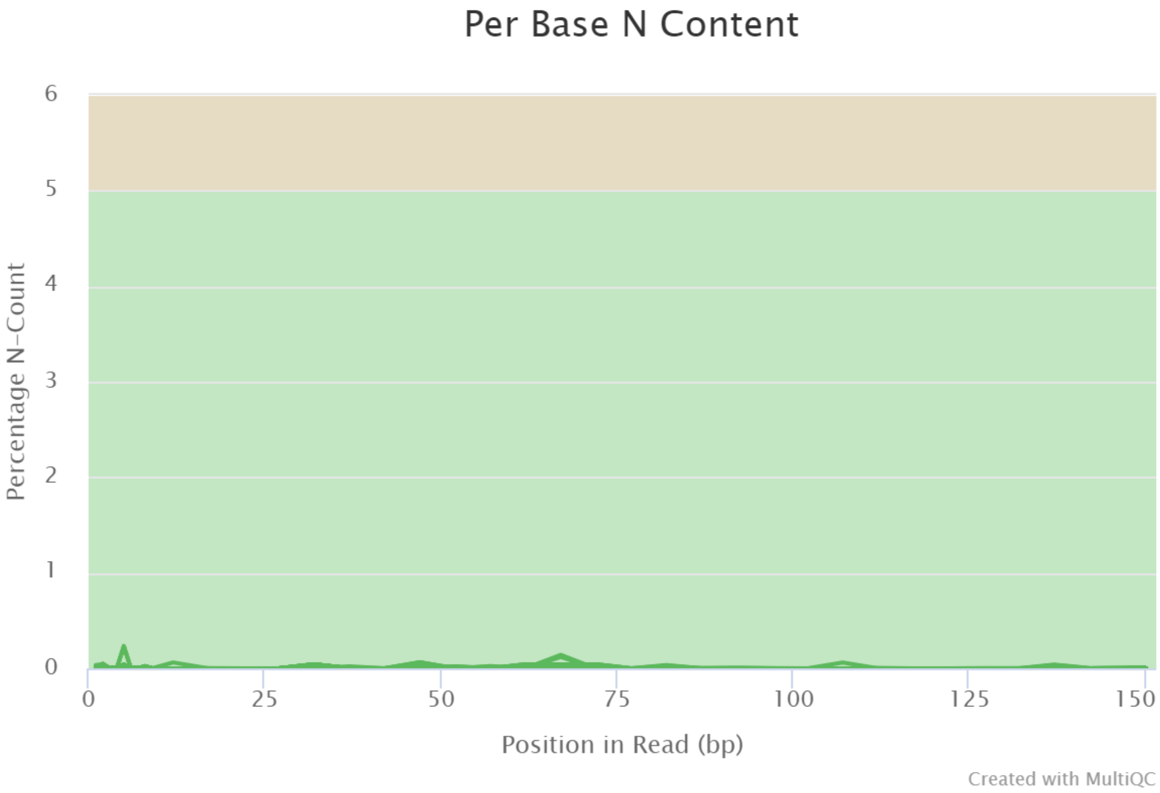


Figure 4: MultiQC output. Per base N content.

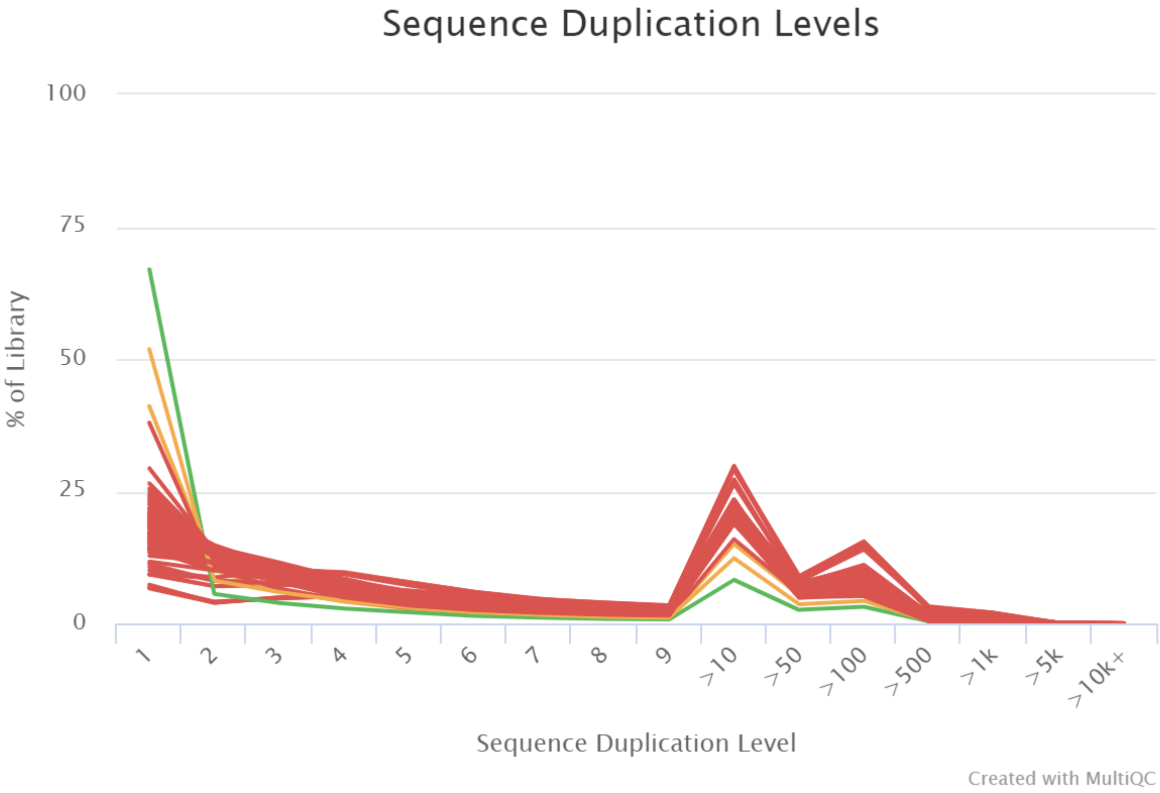


Figure 5: MultiQC output. Sequence duplication levels.

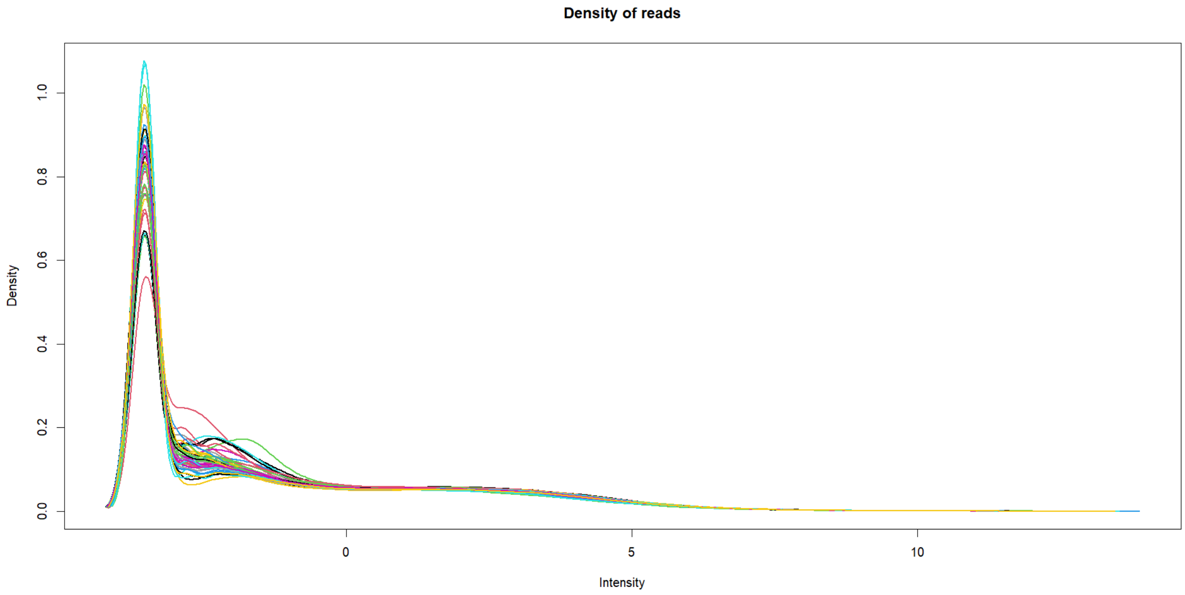
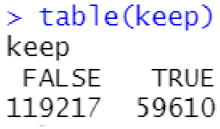


Figure 6: Density of counts before filtering.

Filtered counts with more than 20 reads



```
> table(keep)
keep
 FALSE   TRUE 
119217  59610 
```

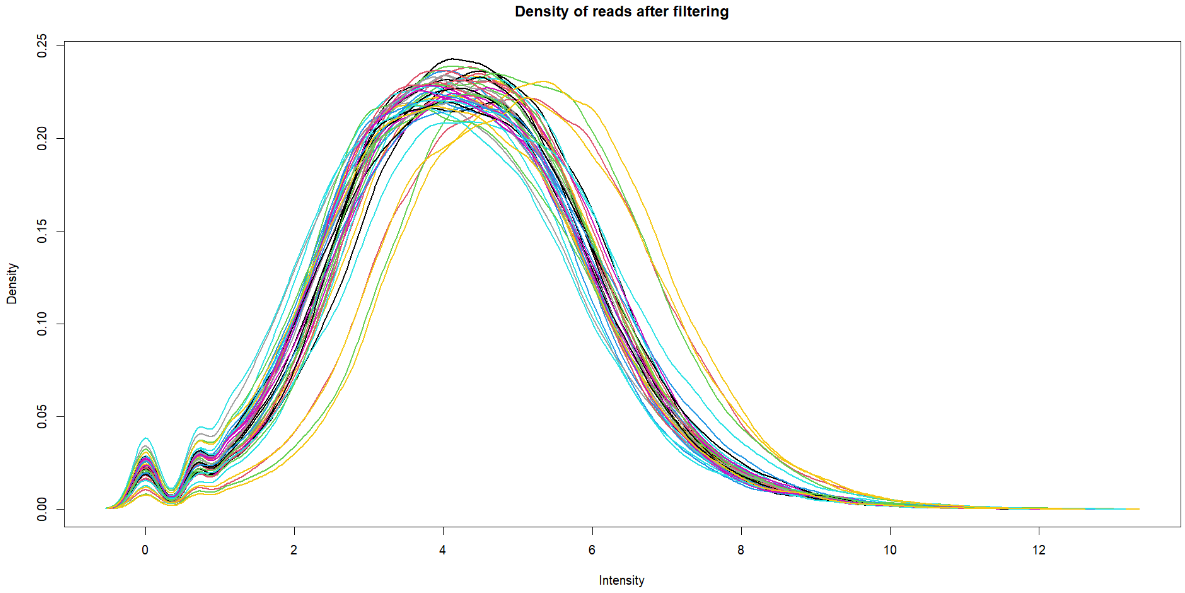
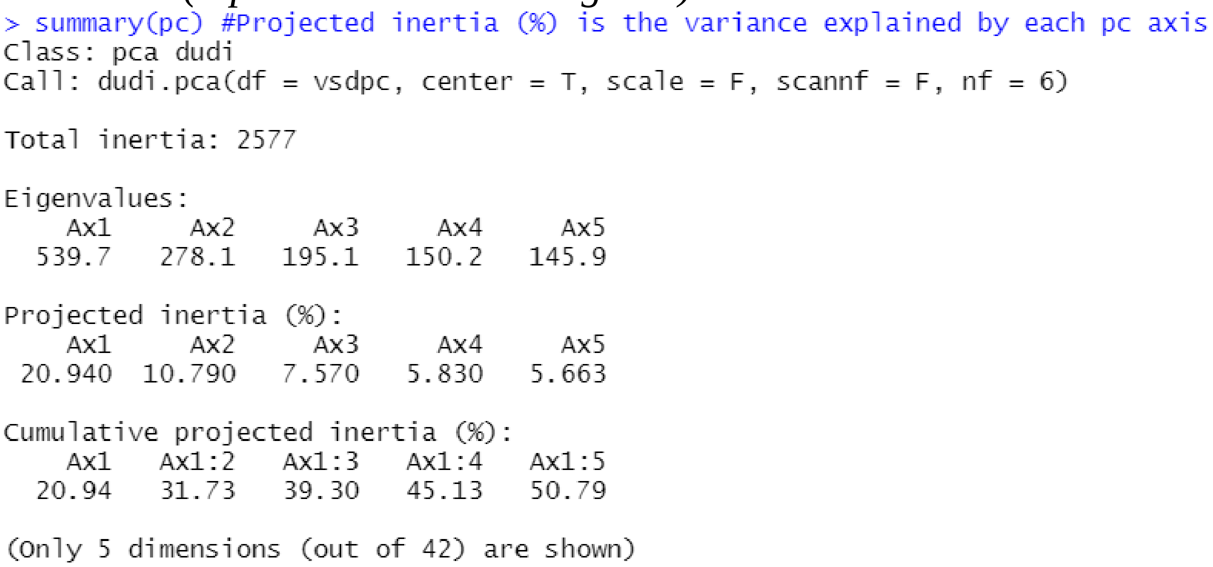


Figure 7: Density of counts after filtering.

PCA stats (top 500 most variable genes)



```
> summary(pc) #Projected inertia (%) is the variance explained by each pc axis
Class: pca dudi
Call: dudi.pca(df = vsdpc, center = T, scale = F, scannf = F, nf = 6)

Total inertia: 2577

Eigenvalues:
    Ax1     Ax2     Ax3     Ax4     Ax5 
  539.7   278.1   195.1   150.2   145.9 

Projected inertia (%):
    Ax1     Ax2     Ax3     Ax4     Ax5 
 20.940  10.790   7.570   5.830   5.663 

Cumulative projected inertia (%):
    Ax1   Ax1:2   Ax1:3   Ax1:4   Ax1:5 
  20.94   31.73   39.30   45.13   50.79 

(Only 5 dimensions (out of 42) are shown)
```

Number of regulated genes

Table 3: number of differentially expressed genes for different contrasts.

	Up	Down	Not significant
MAX1vsC1	1579	1245	56786
MAX3vsC3	570	631	58409
MAX9vsC9	531	482	58597
C9vsC0	438	151	59021
MED9vsMED1	1150	1550	56910
MAX9vsMAX1	1572	2069	55969
C0vsUC	231	106	59273

Evidence codes

Table 3: Evidence codes used to annotate GO terms to the differentially expressed genes. Default Blast2Go values were selected.

Computational analysis evidence codes	
ISS	0.8
ISO	0.8
ISA	0.8
ISM	0.8
IGC	0.7

IBA	0.8
IBD	0.8
IKR	0.8
IRD	0.7
RCA	0.8
Experimental evidence codes	
IDA	1
IPI	1
IMP	1
IGI	1
IEP	1
EXP	1
Author statement evidence codes	
TAS	0.9
NAS	0.8
Curator statement evidence codes	
IC	0.9
ND	0.5
Automatically-assigned evidence codes	
IEA	0.7
Obsolete evidence codes	
NR	0

Appendix II

R scripts

R script 1

```
rm(list=ls())

library(ggplot2)
library(tidyr)
library(dplyr)
library(ggpubr)
library(cowplot)
library(readxl)
library(RColorBrewer)

Stilkbunn <- read_excel("~/MASTER/Resultater/Data/B4K/data_stilkbunn.xlsx")
Stilkbunn$Part <- as.factor(Stilkbunn$Part)
Stilkbunn$Seksjon <- as.factor(Stilkbunn$Seksjon) # ikke som faktor for regresjon
Stilkbunn$Bryozoa <- as.factor(Stilkbunn$Bryozoa)
sapply(Stilkbunn, class)

bunndata <- read_excel("~/MASTER/Resultater/Data/B4K/data_bunn.xlsx")
bunndata$Seksjon <- as.factor(bunndata$Seksjon)
bunndata$Part <- as.factor(bunndata$Part)
bunndata$Bryozoa <- as.factor(bunndata$Bryozoa)
bunndata <-  bunndata %>% drop_na(Ash)

### ANOVA DEL OG SEKSJON OG INDIVID ###
# ANOVA krever Seksjon og del som factor

# jod.ANOVA2 <- aov(Jod~Seksjon/Individ+Part, data=Stilkbunn)
# summary(jod.ANOVA2)

# as.ANOVA2 <- aov(As~Seksjon/Individ+Part, data=Stilkbunn)
# summary(as.ANOVA2)

# br.ANOVA2 <- aov(Br~Seksjon/Individ+Part, data=Stilkbunn)
# summary(br.ANOVA2)
#
# p.ANOVA2 <- aov(P~Seksjon/Individ+Part, data=Stilkbunn)
# summary(p.ANOVA2)
#
# v.ANOVA2 <- aov(V~Seksjon/Individ+Part, data=Stilkbunn)
# summary(v.ANOVA2)
#
# co.ANOVA2 <- aov(Co~Seksjon/Individ+Part, data=Stilkbunn)
# summary(co.ANOVA2)
#
# cd.ANOVA2 <- aov(Cd~Seksjon/Individ+Part, data=Stilkbunn)
# summary(cd.ANOVA2)
```

```
#
# pb.ANOVA2 <- aov(Pb~Seksjon/Individ+Part, data=Stilkbunn)
# summary(pb.ANOVA2)

# Ingen elementer har sign effekt av individ


# SJEKKER COMPLETE MODEL

# jod.ANOVAl <- aov(Jod~Seksjon*Part, data=Stilkbunn)
# summary(jod.ANOVAl)
#
# as.ANOVAl <- aov(As~Seksjon*Part, data=Stilkbunn)
# summary(as.ANOVAl)
#
# br.ANOVAl <- aov(Br~Seksjon*Part, data=Stilkbunn)
# summary(br.ANOVAl)
#
# p.ANOVAl <- aov(P~Seksjon*Part, data=Stilkbunn)
# summary(p.ANOVAl)
#
# v.ANOVAl <- aov(V~Seksjon*Part, data=Stilkbunn)
# summary(v.ANOVAl) # V er eneste element som har sign interaction (*)
#
# co.ANOVAl <- aov(Co~Seksjon*Part, data=Stilkbunn)
# summary(co.ANOVAl)
#
# cd.ANOVAl <- aov(Cd~Seksjon*Part, data=Stilkbunn)
# summary(cd.ANOVAl)
#
# pb.ANOVAl <- aov(Pb~Seksjon*Part, data=Stilkbunn)
# summary(pb.ANOVAl)
#
# # ENDELIG MODELL
#
# jod.ANOVA6 <- aov(Jod~Seksjon+Part, data=Stilkbunn)
# summary(jod.ANOVA6)
#
# as.ANOVA6 <- aov(As~Seksjon+Part, data=Stilkbunn)
# summary(as.ANOVA6)
#
# br.ANOVA6 <- aov(Br~Seksjon+Part, data=Stilkbunn)
# summary(br.ANOVA6)
#
# p.ANOVA6 <- aov(P~Seksjon+Part, data=Stilkbunn)
# summary(p.ANOVA6)
#
# v.ANOVA6 <- aov(V~Seksjon+Part, data=Stilkbunn)
# summary(v.ANOVA6)
#
# co.ANOVA6 <- aov(Co~Seksjon+Part, data=Stilkbunn)
# summary(co.ANOVA6)
#
# cd.ANOVA6 <- aov(Cd~Seksjon+Part, data=Stilkbunn)
# summary(cd.ANOVA6)
#
# pb.ANOVA6 <- aov(Pb~Seksjon+Part, data=Stilkbunn)
# summary(pb.ANOVA6)

# SCATTERPLOTS VEKT
# Seksjon som factor her gir IKKE regresjonslinje
# NUMERISK funker med regresjonslinje

# SEKSJON VS VEKT

seksjon <- ggplot(bunndata, aes(x=Seksjon, y=Weight)) +
  geom_point(aes(), size=6) +
  # geom_smooth(method="", se=F) +
  labs(title = "Thallus size",
       x = "Rope section",
       y = "Weight (g)") +
  theme(text = element_text(size = 20),
        axis.text = element_text(face="bold", color="black", size=18),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        plot.margin = unit(c(2,1,2,1), "cm"))

ash <- ggplot(bunndata, aes(x=Ash, y=Weight)) +
  geom_point(aes(), size=6) +
  # geom_smooth(method="", se=F) +
  labs(title = "Thallus size",
       x = "% ash content",
       y = "Weight (g)") +
  theme(text = element_text(size = 20),
        axis.text = element_text(face="bold", color="black", size=18),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        plot.margin = unit(c(2,1,2,1), "cm"))

plot_grid(seksjon, ash)

# ELEMENTS 1

iodine <- ggplot(Stilkbunn, aes(x=Seksjon, y=Jod)) +
  geom_point(aes(col=Part, shape=Part), size=6) +
  geom_smooth(method="", se=F) +
  labs(title = "Iodine",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")
```

```
bromine <- ggplot(Stilkbunn, aes(x=Seksjon, y=Br)) +
  geom_point(aes(col=Part, shape=Part), size=6) +
  geom_smooth(method="lm", se=F) +
  labs(title = "Bromine",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        axis.title.y = element_blank(),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

bromine2 <- ggplot(Stilkbunn, aes(x=Seksjon, y=Br)) +
  geom_point(aes(col=Part, shape=Part), size=6) +
  geom_smooth(method="lm", se=F) +
  labs(title = "Bromine",
       x = "Rope section",
       y = "mg/kg",
       col = "Thallus part",
       shape = "Thallus part") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        axis.title.y = element_blank(),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.direction = "vertical")
legend <- cowplot::get_legend(bromine2)
# grid.newpage()
# grid.draw(legend)

arsenic <- ggplot(Stilkbunn, aes(x=Seksjon, y=As)) +
  geom_point(aes(col=Part, shape=Part), size=6) +
  labs(title = "Arsenic",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

cadmium <- ggplot(Stilkbunn, aes(x=Seksjon, y=Cd)) +
  geom_point(aes(col=Part, shape=Part), size=6) +
  geom_smooth(method="lm", se=F) +
  labs(title = "Cadmium",
       x = "Section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        axis.title.y = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

lead <- ggplot(Stilkbunn, aes(x=Seksjon, y=Pb)) +
  geom_point(aes(col=Part, shape=Part), size=6) +
  geom_smooth(method="lm", se=F) +
  labs(title = "Lead",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        axis.title.y = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

vanadium <- ggplot(Stilkbunn, aes(x=Seksjon, y=V)) +
  geom_point(aes(col=Part, shape=Part), size=6) +
  geom_smooth(method="lm", se=F) +
  labs(title = "Vanadium",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

phosphorus <- ggplot(Stilkbunn, aes(x=Seksjon, y=P)) +
  geom_point(aes(col=Part, shape=Part), size=6) +
  geom_smooth(method="lm", se=F) +
  labs(title = "Phosphorus",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        axis.title.y = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

cobalt <- ggplot(Stilkbunn, aes(x=Seksjon, y=Co)) +
  geom_point(aes(col=Part, shape=Part), size=6) +
  geom_smooth(method="", se=F) +
  labs(title = "Cobalt",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        axis.title.y = element_blank(),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

plots <- align_plots(iodine, bromine, legend, align = 'h', axis = 'r')

top_row <- plot_grid(plots[[1]], plots[[2]], plots[[3]],
                     #labels = "", label_size = 12,
                     ncol = 3)

plots2 <- align_plots(arsenic, cadmium, lead,
                     align = 'h', axis = 'l')

middle_row <- plot_grid(plots2[[1]], plots2[[2]], plots2[[3]],
                        #labels = c("C", "D", "E"), label_size = 12,
                        ncol = 3)

plots3 <- align_plots(vanadium, phosphorus, cobalt,
                     align = 'h', axis = 'l')

bottom_row <- plot_grid(plots3[[1]], plots3[[2]], plots3[[3]],
                        #labels = c("F", "G", "H"), label_size = 12,
                        ncol = 3)

cowplot::plot_grid(top_row, middle_row, bottom_row,
                   ncol = 1, nrow = 3)


# ELEMENTS 2 ASH

iodine <- ggplot(bunndata, aes(x=Ash, y=Jod)) +
  geom_point(aes(col=Weight), size=6) +
  labs(title = "Iodine",
       x = "% ash content",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

iodine2 <- ggplot(bunndata, aes(x=Ash, y=Jod)) +
  geom_point(aes(col=Weight), size=6) +
  labs(title = "Iodine",
       x = "Section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.direction = "vertical")

bromine <- ggplot(bunndata, aes(x=Ash, y=Br)) +
  geom_point(aes(col=Weight), size=6) +
  geom_smooth(method="lm", se=F) +
  labs(title = "Bromine",
       x = "% ash content",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        axis.title.y = element_blank(),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

arsenic <- ggplot(bunndata, aes(x=Ash, y=As)) +
  geom_point(aes(col=Weight), size=6) +
  labs(title = "Arsenic",
       x = "% ash content",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

cadmium <- ggplot(bunndata, aes(x=Ash, y=Cd)) +
  geom_point(aes(col=Weight), size=6) +
  labs(title = "Cadmium",
       x = "% ash content",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        axis.title.y = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

lead <- ggplot(bunndata, aes(x=Ash, y=Pb)) +
  geom_point(aes(col=Weight), size=6) +
  labs(title = "Lead",
       x = "% ash content",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        axis.title.y = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

vanadium <- ggplot(bunndata, aes(x=Ash, y=V)) +
  geom_point(aes(col=Weight), size=6) +
  labs(title = "Vanadium",
       x = "% ash content",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

phosphorus <- ggplot(bunndata, aes(x=Ash, y=P)) +
  geom_point(aes(col=Weight), size=6) +
  geom_smooth(method="lm", se=F) +
  labs(title = "Phosphorus",
       x = "% ash content",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        axis.title.y = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

cobalt <- ggplot(bunndata, aes(x=Ash, y=Co)) +
  geom_point(aes(col=Weight), size=6) +
  labs(title = "Cobalt",
       x = "% ash content",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        axis.title.y = element_blank(),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

iodine <- iodine + scale_color_gradient(low = "aquamarine", high = "darkblue")
iodine2 <- iodine2 + scale_color_gradient(low="aquamarine", high="darkblue")
legend <- cowplot::get_legend(iodine2)
bromine <- bromine + scale_color_gradient(low="aquamarine", high="darkblue")
arsenic <- arsenic + scale_color_gradient(low="aquamarine", high="darkblue")
cadmium <- cadmium + scale_color_gradient(low="aquamarine", high="darkblue")
lead <- lead + scale_color_gradient(low="aquamarine", high="darkblue")
vanadium <- vanadium + scale_color_gradient(low="aquamarine", high="darkblue")
phosphorus <- phosphorus + scale_color_gradient(low="aquamarine", high="darkblue")
cobalt <- cobalt + scale_color_gradient(low="aquamarine", high="darkblue")

plots <- align_plots(iodine, bromine, legend, align = 'h', axis = 'r')

top_row <- plot_grid(plots[[1]], plots[[2]], plots[[3]],
                     #labels = "", label_size = 12,
                     ncol = 3)

plots2 <- align_plots(arsenic, cadmium, lead,
                      align = 'h', axis = 'l')

middle_row <- plot_grid(plots2[[1]], plots2[[2]], plots2[[3]],
                        #labels = c("C", "D", "E"), label_size = 12,
                        ncol = 3)

plots3 <- align_plots(vanadium, phosphorus, cobalt,
                      align = 'h', axis = 'l')

bottom_row <- plot_grid(plots3[[1]], plots3[[2]], plots3[[3]],
                        #labels = c("F", "G", "H"), label_size = 12,
                        ncol = 3)

cowplot::plot_grid(top_row, middle_row, bottom_row,
                   ncol = 1, nrow = 3)

ash_anova1 <- aov(Jod ~ Ash, data = bunndata)
summary(ash_anova1)
ash_anova2 <- aov(As ~ Ash, data = bunndata)
summary(ash_anova2)
ash_anova3 <- aov(Br ~ Ash, data = bunndata)
summary(ash_anova3)
ash_anova4 <- aov(P ~ Ash, data = bunndata)
summary(ash_anova4)
ash_anova5 <- aov(V ~ Ash, data = bunndata)
summary(ash_anova5)
ash_anova6 <- aov(Co ~ Ash, data = bunndata)
summary(ash_anova6)
ash_anova7 <- aov(Cd ~ Ash, data = bunndata)
```

```
summary(ash_anova7)
ash_anova8 <- aov(Pb ~ Ash, data = bunndata)
summary(ash_anova8)

# ELEMENTS 3 WEIGHT

iodine <- ggplot(bunndata, aes(x=Jod, y=Weight)) +
  geom_point(aes(col=Bryozoa), size=6) +
  labs(title = "Weight",
       x = "Iodine (mg/kg)",
       y = "Weight (g)") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

iodine2 <- ggplot(bunndata, aes(x=Jod, y=Weight)) +
  geom_point(aes(col=Bryozoa), size=6) +
  labs(title = "Weight",
       x = "Iodine (mg/kg)",
       y = "Weight (g)") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.direction = "vertical")
legend <- cowplot::get_legend(iodine2)

bromine <- ggplot(bunndata, aes(x=Br, y=Weight)) +
  geom_point(aes(col=Bryozoa), size=6) +
  labs(title = "Weight",
       x = "Bromine (mg/kg)",
       y = "Weight (g)") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        axis.title.y = element_blank(),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

arsenic <- ggplot(bunndata, aes(x=As, y=Weight)) +
  geom_point(aes(col=Bryozoa), size=6) +
  labs(title = "Weight",
       x = "Arsenic (mg/kg)",
       y = "Weight (g)") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

cadmium <- ggplot(bunndata, aes(x=Cd, y=Weight)) +
  geom_point(aes(col=Bryozoa), size=6) +
  labs(title = "Weight",
       x = "Cadmium (mg/kg)",
       y = "Weight (g)") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        axis.title.y = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

lead <- ggplot(bunndata, aes(x=Pb, y=Weight)) +
  geom_point(aes(col=Bryozoa), size=6) +
  labs(title = "Weight",
       x = "Lead (mg/kg)",
       y = "Weight (g)") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        axis.title.y = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

vanadium <- ggplot(bunndata, aes(x=V, y=Weight)) +
  geom_point(aes(col=Bryozoa), size=6) +
  labs(title = "Weight",
       x = "Vanadium (mg/kg)",
       y = "Weight (g)") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

phosphorus <- ggplot(bunndata, aes(x=P, y=Weight)) +
  geom_point(aes(col=Bryozoa), size=6) +
  geom_smooth(method="lm", se=F) +
  labs(title = "Weight",
      x = "Phosphorus (mg/kg)",
      y = "Weight (g)") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        axis.title.y = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

cobalt <- ggplot(bunndata, aes(x=Co, y=Weight)) +
  geom_point(aes(col=Bryozoa), size=6) +
  labs(title = "Weight",
      x = "Cobalt (mg/kg)",
      y = "Weight (g)") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        axis.title.y = element_blank(),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

iodine <- iodine + scale_color_gradient(low = "#E69F00", high = "#F0E442")
bromine <- bromine + scale_color_gradient(low="aquamarine", high="darkblue")
bromine2 <- bromine2 + scale_color_gradient(low="aquamarine", high="darkblue")
arsenic <- arsenic + scale_color_gradient(low="aquamarine", high="darkblue")
cadmium <- cadmium + scale_color_gradient(low="aquamarine", high="darkblue")
lead <- lead + scale_color_gradient(low="aquamarine", high="darkblue")
vanadium <- vanadium + scale_color_gradient(low="aquamarine", high="darkblue")
phosphorus <- phosphorus + scale_color_gradient(low="aquamarine", high="darkblue")
cobalt <- cobalt + scale_color_gradient(low="aquamarine", high="darkblue")

plots <- align_plots(iodine, bromine, legend, align = 'h', axis = 'r')

top_row <- plot_grid(plots[[1]], plots[[2]], plots[[3]],
                    #labels = "", label_size = 12,
                    ncol = 3)

plots2 <- align_plots(arsenic, cadmium, lead,
                     align = 'h', axis = 'l')

middle_row <- plot_grid(plots2[[1]], plots2[[2]], plots2[[3]],
                       #labels = c("C", "D", "E"), label_size = 12,
                       ncol = 3)

plots3 <- align_plots(vanadium, phosphorus, cobalt,
                     align = 'h', axis = 'l')

bottom_row <- plot_grid(plots3[[1]], plots3[[2]], plots3[[3]],
                       #labels = c("F", "G", "H"), label_size = 12,
                       ncol = 3)
```

```
cowplot::plot_grid(top_row, middle_row, bottom_row,
                  ncol = 1, nrow = 3)

weight_anova1 <- aov(Weight ~ Jod, data = bunndata)
summary(weight_anova1)
weight_anova2 <- aov(Weight ~ Br, data = bunndata)
summary(weight_anova2)
weight_anova3 <- aov(Weight ~ As, data = bunndata)
summary(weight_anova3)
weight_anova4 <- aov(Weight ~ Cd, data = bunndata)
summary(weight_anova4)
weight_anova5 <- aov(Weight ~ Pb, data = bunndata)
summary(weight_anova5)
weight_anova6 <- aov(Weight ~ V, data = bunndata)
summary(weight_anova6)
weight_anova7 <- aov(Weight ~ P, data = bunndata)
summary(weight_anova7)
weight_anova8 <- aov(Weight ~ Co, data = bunndata)
summary(weight_anova8)

section_anova1 <- aov(Jod ~ Seksjon*Weight*Ash, data = bunndata)
summary(section_anova1)
section_anova2 <- aov(Br ~ Seksjon+Weight+Ash, data = bunndata)
summary(section_anova2)
section_anova3 <- aov(As ~ Seksjon+Weight+Ash, data = bunndata)
summary(section_anova3)
section_anova4 <- aov(Cd ~ Seksjon+Weight+Ash, data = bunndata)
summary(section_anova4)
section_anova5 <- aov(Pb ~ Seksjon+Weight+Ash, data = bunndata)
summary(section_anova5)
section_anova6 <- aov(V ~ Seksjon+Weight+Ash, data = bunndata)
summary(section_anova6)
section_anova7 <- aov(P ~ Seksjon+Weight+Ash, data = bunndata)
summary(section_anova7)
section_anova8 <- aov(Co ~Seksjon+Weight+Ash, data = bunndata)
summary(section_anova8)

bryo_anova1 <- aov(Jod ~ Bryozoa, data = Stilkbunn)
summary(bryo_anova1)
bryo_anova2 <- aov(As ~ Bryozoa, data = Stilkbunn)
summary(bryo_anova2)
bryo_anova3 <- aov(Br ~ Bryozoa, data = Stilkbunn)
summary(bryo_anova3)
bryo_anova4 <- aov(P ~ Bryozoa, data = Stilkbunn)
summary(bryo_anova4)
bryo_anova5 <- aov(V ~ Bryozoa, data = Stilkbunn)
summary(bryo_anova5)
bryo_anova6 <- aov(Co ~ Bryozoa, data = Stilkbunn)
summary(bryo_anova6)
bryo_anova7 <- aov(Cd ~ Bryozoa, data = Stilkbunn)
summary(bryo_anova7)
bryo_anova8 <- aov(Pb ~ Bryozoa, data = Stilkbunn)
summary(bryo_anova8)

bryo_anova9 <- aov(Bryozoa ~ Part, data = Stilkbunn)
summary(bryo_anova9)
```

```
weight_anova9 <- aov(Weight ~ Seksjon, data = bunndata)
summary(weight_anova9)

# ELEMENTS 4

iodine <- ggplot(bunndata, aes(x=Seksjon, y=Jod)) +
  geom_point(aes(col=Weight), size=6) +
  geom_smooth(method="lm", se=F) +
  labs(title = "Iodine",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

bromine <- ggplot(bunndata, aes(x=Seksjon, y=Br)) +
  geom_point(aes(col=Weight), size=6) +
  labs(title = "Bromine",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        axis.title.y = element_blank(),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

bromine2 <- ggplot(bunndata, aes(x=Seksjon, y=Br)) +
  geom_point(aes(col=Weight), size=6) +
  labs(title = "Bromine",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        axis.title.y = element_blank(),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.direction = "vertical")
legend <- cowplot::get_legend(bromine2)
# grid.newpage()
# grid.draw(legend)

arsenic <- ggplot(bunndata, aes(x=Seksjon, y=As)) +
  geom_point(aes(col=Weight), size=6) +
  geom_smooth(method="lm", se=F) +
  labs(title = "Arsenic",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

cadmium <- ggplot(bunndata, aes(x=Seksjon, y=Cd)) +
  geom_point(aes(col=Weight), size=6) +
  labs(title = "Cadmium",
       x = "Section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        axis.title.y = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

lead <- ggplot(bunndata, aes(x=Seksjon, y=Pb)) +
  geom_point(aes(col=Weight), size=6) +
  labs(title = "Lead",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        axis.title.y = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

vanadium <- ggplot(bunndata, aes(x=Seksjon, y=V)) +
  geom_point(aes(col=Weight), size=6) +
  labs(title = "Vanadium",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")
```

```
phosphorus <- ggplot(bunndata, aes(x=Seksjon, y=P)) +
  geom_point(aes(col=Weight), size=6) +
  labs(title = "Phosphorus",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        axis.title.y = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")


cobalt <- ggplot(bunndata, aes(x=Seksjon, y=Co)) +
  geom_point(aes(col=Weight), size=6) +
  labs(title = "Cobalt",
       x = "Rope section",
       y = "mg/kg") +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        axis.title.y = element_blank(),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")


iodine <- iodine + scale_color_gradient(low="aquamarine", high="darkblue")
bromine <- bromine + scale_color_gradient(low="aquamarine", high="darkblue")
bromine2 <- bromine2 + scale_color_gradient(low="aquamarine", high="darkblue")
arsenic <- arsenic + scale_color_gradient(low="aquamarine", high="darkblue")
cadmium <- cadmium + scale_color_gradient(low="aquamarine", high="darkblue")
lead <- lead + scale_color_gradient(low="aquamarine", high="darkblue")
vanadium <- vanadium + scale_color_gradient(low="aquamarine", high="darkblue")
phosphorus <- phosphorus + scale_color_gradient(low="aquamarine", high="darkblue")
cobalt <- cobalt + scale_color_gradient(low="aquamarine", high="darkblue")

plots <- align_plots(iodine, bromine, legend, align = 'h', axis = 'r')

top_row <- plot_grid(plots[[1]], plots[[2]], plots[[3]],
                     #labels = "", label_size = 12,
                     ncol = 3)

plots2 <- align_plots(arsenic, cadmium, lead,
                      align = 'h', axis = 'l')

middle_row <- plot_grid(plots2[[1]], plots2[[2]], plots2[[3]],
                        #labels = c("C", "D", "E"), label_size = 12,
                        ncol = 3)
```

```
plots3 <- align_plots(vanadium, phosphorus, cobalt,
                      align = 'h', axis = 'l')

bottom_row <- plot_grid(plots3[[1]], plots3[[2]], plots3[[3]],
                        #labels = c("F", "G", "H"), label_size = 12,
                        ncol = 3)

cowplot::plot_grid(top_row, middle_row, bottom_row,
                   ncol = 1, nrow = 3)

section_anova1 <- aov(Jod ~ Seksjon, data = bunndata)
summary(section_anova1)
section_anova2 <- aov(Br ~ Seksjon, data = bunndata)
summary(section_anova2)
section_anova3 <- aov(As ~ Seksjon, data = bunndata)
summary(section_anova3)
section_anova4 <- aov(Cd ~ Seksjon, data = bunndata)
summary(section_anova4)
section_anova5 <- aov(Pb ~ Seksjon, data = bunndata)
summary(section_anova5)
section_anova6 <- aov(V ~ Seksjon, data = bunndata)
summary(section_anova6)
section_anova7 <- aov(P ~ Seksjon, data = bunndata)
summary(section_anova7)
section_anova8 <- aov(Co ~Seksjon, data = bunndata)
summary(section_anova8)
```

R script 2

```
rm(list=ls())

library(ggplot2)
library(tidyr)
library(dplyr)
library(ggpubr)
library(ggfortify)
library(Hmisc)
library(kableExtra)
library(readxl)
library(ggfortify)

safepca <- read_excel("~/MASTER/Resultater/Data/data_safekelp.xlsx")
safepca$Light <- as.factor(safepca$Light)
safepca$Time <- as.factor(safepca$Time)
sapply(safepca, class)
View(safepca)
str(safepca)

# PCA 2

pca2 <- prcomp(safepca[-c(1:8),8:15], scale = TRUE)
summary(pca2)

# safepca2 <- cbind(safepca, pca2$x[,1:6])
# View(safepca2)

# PLOTS

plot(pca2, type="l")

# biplot(pca2, scale=0)

autoplot(pca2, data =safepca[-c(1:8),], size= 6,
         shape = "Time", colour= "Light",
         loadings= T, label = F,
         loadings.label = T, loadings.label.size  = 6) +
  labs(title="PCA Time and Light") +
  theme(text = element_text(size = 20),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5))
```

R script 3

```
rm(list=ls())

library(ggplot2)
library(tidyr)
library(dplyr)
library(ggpubr)
library(readxl)
library(cowplot)
library(RColorBrewer)

Safedata <- read_excel("~/MASTER/Resultater/Data/data_safekelp.xlsx")
Safedata$Light <- as.factor(Safedata$Light)    # best med time som integer?
Safedata$Time <- as.integer(Safedata$Time)      # og light som factor
Safedata$Replikat <- as.integer(Safedata$Replikat)
Safedata$Cut <- as.factor(Safedata$Cut)
sapply(Safedata, class)
View(Safedata)



# Safedata$Time <- as.factor(Safedata$Time)
# Safedata$Light <- as.integer(Safedata$Light)
#
# k <- levels(Safedata$Light)[Safedata$Light]
# Safedata$Light <- as.integer(k)
#
# l <- levels(Safedata$Time)[Safedata$Time]
# Safedata$Time <- as.integer(l)
#
# str(Safedata)
# glimpse(Safedata)
# summary(Safedata)
# dim(Safedata)

# GG SCATTERPLOTS

# JOD


# ELEMENTS 1

iodine <- ggplot(Safedata, aes(x=Time, y=Jod)) +
  geom_point(aes(col=Light), size=6) +
  labs(title = "Iodine",
       x = "Time (days)",
       y = "mg/kg") +
  scale_x_continuous(breaks= c(0,1,3,9)) +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

bromine <- ggplot(Safedata, aes(x=Time, y=Br)) +
  geom_point(aes(col=Light), size=6) +
  labs(title = "Bromine",
       x = "Time (days)",
       y = "mg/kg") +
  scale_x_continuous(breaks= c(0,1,3,9)) +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        axis.title.y = element_blank(),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

bromine2 <- ggplot(Safedata, aes(x=Time, y=Br)) +
  geom_point(aes(col=Light), size=6) +
  labs(title = "Bromine",
       x = "Time (days)",
       y = "mg/kg",
       col = "Light") +
  scale_x_continuous(breaks= c(0,1,3,9)) +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        axis.title.y = element_blank(),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.direction = "horizontal")
legend <- cowplot::get_legend(bromine2)
# grid.newpage()
# grid.draw(legend)

arsenic <- ggplot(Safedata, aes(x=Time, y=As)) +
  geom_point(aes(col=Light), size=6) +
  labs(title = "Arsenic",
       x = "Time (days)",
       y = "mg/kg") +
  scale_x_continuous(breaks= c(0,1,3,9)) +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

cadmium <- ggplot(Safedata, aes(x=Time, y=Cd)) +
  geom_point(aes(col=Light), size=6) +
  labs(title = "Cadmium",
       x = "Time (days)",
       y = "mg/kg") +
  scale_x_continuous(breaks= c(0,1,3,9)) +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        axis.title.y = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

lead <- ggplot(Safedata, aes(x=Time, y=Pb)) +
  geom_point(aes(col=Light), size=6) +
  labs(title = "Lead",
       x = "Time (days)",
       y = "mg/kg") +
  scale_x_continuous(breaks= c(0,1,3,9)) +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

vanadium <- ggplot(Safedata, aes(x=Time, y=V)) +
  geom_point(aes(col=Light), size=6) +
  labs(title = "Vanadium",
       x = "Time (days)",
       y = "mg/kg") +
  scale_x_continuous(breaks= c(0,1,3,9)) +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        axis.title.y = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

phosphorus <- ggplot(Safedata, aes(x=Time, y=P)) +
  geom_point(aes(col=Light), size=6) +
  labs(title = "Phosphorus",
       x = "Time (days)",
       y = "mg/kg") +
  scale_x_continuous(breaks= c(0,1,3,9)) +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")

cobalt <- ggplot(Safedata, aes(x=Time, y=Co)) +
  geom_point(aes(col=Light), size=6) +
  labs(title = "Cobalt",
       x = "Time (days)",
       y = "mg/kg") +
  scale_x_continuous(breaks= c(0,1,3,9)) +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        axis.title.y = element_blank(),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")




cowplot::plot_grid(legend, middle_row, bottom_row,
                   ncol = 1, nrow = 3)







plots <- align_plots(iodine, bromine, align = 'h', axis = 'r')
```

```
top_row <- plot_grid(plots[[1]], plots[[2]],
                    #labels = "", label_size = 12,
                    ncol = 2)

plots2 <- align_plots(arsenic, cadmium,
                     align = 'h', axis = 'l')

middle_row <- plot_grid(plots2[[1]], plots2[[2]],
                       #labels = c("C", "D", "E"), label_size = 12,
                       ncol = 2)

cowplot::plot_grid(top_row, middle_row,
                  ncol=1, nrow = 2)



plots3 <- align_plots(lead, vanadium,
                     align = 'h', axis = 'l')

bottom_row <- plot_grid(plots3[[1]], plots3[[2]],
                       #labels = c("F", "G", "H"), label_size = 12,
                       ncol = 2)

plots4 <- align_plots(phosphorus, cobalt,
                     align = 'h', axis = 'l')

bottoms <- plot_grid(plots4[[1]], plots4[[2]],
                       #labels = c("F", "G", "H"), label_size = 12,
                       ncol = 2)

cowplot::plot_grid(bottom_row, bottoms,
                  ncol = 1, nrow = 2)



# ANOVA HEL ELLER SKJAERT

cutANOVA1 <- aov(Jod~Cut, data=Safedata[1:8,])
summary(cutANOVA1)

cutANOVA2 <- aov(As~Cut, data=Safedata[1:8,])
summary(cutANOVA2)

cutANOVA3 <- aov(Br~Cut, data=Safedata[1:8,])
summary(cutANOVA3)

cutANOVA4 <- aov(P~Cut, data=Safedata[1:8,])
summary(cutANOVA4)

cutANOVA5 <- aov(V~Cut, data=Safedata[1:8,])
summary(cutANOVA5)

cutANOVA6 <- aov(Co~Cut, data=Safedata[1:8,])
summary(cutANOVA6)
```

```
cutANOVA7 <- aov(Cd~Cut, data=Safedata[1:8,])
summary(cutANOVA7)

cutANOVA8 <- aov(Pb~Cut, data=Safedata[1:8,])
summary(cutANOVA8)

# ANOVA 4 TIDSPUNKT BARE 40

jod.ANOVA4 <- aov(Jod~Time, data=Safedata[c(1:20),])
summary(jod.ANOVA4)

as.ANOVA4 <- aov(As~Time, data=Safedata[c(1:20),])
summary(as.ANOVA4)

br.ANOVA4 <- aov(Br~Time, data=Safedata[c(1:20),])
summary(br.ANOVA4)

p.ANOVA4 <- aov(P~Time, data=Safedata[c(1:20),])
summary(p.ANOVA4)

v.ANOVA4 <- aov(V~Time, data=Safedata[c(1:20),])
summary(v.ANOVA4)

co.ANOVA4 <- aov(Co~Time, data=Safedata[c(1:20),])
summary(co.ANOVA4)

cd.ANOVA4 <- aov(Cd~Time, data=Safedata[c(1:20),])
summary(cd.ANOVA4)

pb.ANOVA4 <- aov(Pb~Time, data=Safedata[c(1:20),])
summary(pb.ANOVA4)

# ANOVA TID OG LYS

jod.ANOVA5 <- aov(Jod~Time*Light, data=Safedata[-c(1:8),])
summary(jod.ANOVA5)

as.ANOVA5 <- aov(As~Time*Light, data=Safedata[-c(1:8),])
summary(as.ANOVA5)

br.ANOVA5 <- aov(Br~Time*Light, data=Safedata[-c(1:8),])
summary(br.ANOVA5)

p.ANOVA5 <- aov(P~Time*Light, data=Safedata[-c(1:8),])
summary(p.ANOVA5)

v.ANOVA5 <- aov(V~Time*Light, data=Safedata[-c(1:8),])
summary(v.ANOVA5)

co.ANOVA5 <- aov(Co~Time*Light, data=Safedata[-c(1:8),])
summary(co.ANOVA5)

cd.ANOVA5 <- aov(Cd~Time*Light, data=Safedata[-c(1:8),])
summary(cd.ANOVA5)

pb.ANOVA5 <- aov(Pb~Time*Light, data=Safedata[-c(1:8),])
summary(pb.ANOVA5)
```

```
# EKSTRA

jod.ANOVA7 <- aov(Jod~Time, data=Safedata[-c(1:8),])
summary(jod.ANOVA7)

jod.ANOVA8 <- aov(Jod~Light, data=Safedata[-c(1:8),])
summary(jod.ANOVA8)

jod.ANOVA9 <- aov(Jod~Time*Light, data=Safedata[-c(1:8),])
summary(jod.ANOVA9)




iodine <- ggplot(Safedata[1:8,], aes(x=Time, y=Jod)) +
  geom_point(aes(), size=6) +
  labs(title = "Iodine",
       x = "Time (days)",
       y = "mg/kg") +
  scale_x_continuous(breaks= c(0,1,3,9)) +
  theme(text = element_text(size = 25),
        axis.text = element_text(face="bold", color="black", size=21),
        panel.border = element_rect(size = 2, linetype = "solid", fill = NA),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        plot.title=element_text(hjust=0.5),
        legend.position = "none")
```

R script 1

```
# Safekelp RNA-seq analysis

rm(list=ls())

library(tidyverse)
library(data.table)
library(waldo)
library(readxl)
library(edgeR)
library(DESeq2)
library(adegenet)
library(RColorBrewer)
library(adegenet)
library(ade4)
library(factoextra)
library(UpSetR)
library(ggvenn)
library(VennDetail)
library(affy)
library(dplyr)
library(tibble)

setwd("C:/Users/simon/Prosjekter/RNA_seq/r_analysis")

#loding counts matrix file
dta <- read.table("counts.txt", header = T)
dim(dta)
names(dta)

#creating metadata for the counts
mdta <- data.frame(
  id=c(
    'K1','K2','K3','K4','K5','K6','K7','K8','K10','K11','K12','K13','K14','K15','K16','K17',
    'K18','K19','K20','K21','K22','K23','K24','K25','K26','K27','K28','K29','K30','K31',
    'K32','K33','K34','K35','K36','K37','K38','K39','K40','K41','K42','K43','K44'),
  rep=c(
    1,1,1,1,2,2,2,2,3,3,3,4,4,4,4,5,5,5,5,6,6,6,6,7,7,7,7,
    8,8,8,8,9,9,9,9,10,10,10,10,11,11,11,11),
  Light=c(
    'MIN','MIN','MIN','MIN','UC','UC','UC','UC','MIN','MIN','MIN','MIN','MIN','MIN','MIN','MIN',
    'MIN','MIN','MIN','MED','MED','MED','MED','MED','MED','MED','MED','MED','MED','MED','MED',
    'MAX','MAX','MAX','MAX','MAX','MAX','MAX','MAX','MAX','MAX','MAX','MAX'),
  Days=c(
    'C0','C0','C0','C0','C0','C0','C0','C0','D1','D1','D1','D3','D3','D3','D3',
    'D9','D9','D9',
    'D9','D1','D1','D1','D1','D3','D3','D3','D3','D9','D9','D9','D9','D1','D1','D1','D1',
    'D3','D3','D3','D3','D9','D9','D9','D9'),
  group2=c(
    'C0','C0','C0','C0','UC','UC','UC','UC','C1','C1','C1','C3','C3','C3',
    'C3','C9','C9','C9','C9','MED1','MED1','MED1','MED1','MED3',
    'MED3','MED3','MED3','MED9','MED9','MED9','MED9','MAX1','MAX1','MAX1',
    'MAX1','MAX3','MAX3','MAX3','MAX3','MAX9','MAX9','MAX9','MAX9'),
  treatment2=c(
    'C','C','C','C','C','C','C','C','C','C','C','C','C','C','C','C',
    'C','C','C',
    'T','T','T','T','T','T','T','T','T','T','T','T','T','T','T','T',
    'T','T','T','T','T','T','T','T'),
  stringsAsFactors = TRUE)

rownames(mdta) <- mdta$id

##checking the order of samples in metadata and count data
waldo::compare(
  colnames(dta),
  rownames(mdta))

#preparing DEGlist
FtDEG <- DGEList(
  counts = dta,
  samples = mdta,
  remove.zeros = T)
dim(FtDEG)

##plotting distribution of counts
# plotDensities(
#   cpm(
#     FtDEG$counts,
#     log = T),
#   # col = c("red", "blue", "green", "yellow",
#   #         "pink", "orange","black", "gray"),
#   legend = F,
#   main = "Density of reads")

##Filtering transcripts with low read counts
keep <- filterByExpr(
  FtDEG,
  min.count=20, ## BYTTE TIL 30?
  group = FtDEG$samples$group2)

table(keep)

FtDEG <- FtDEG[keep, ,keep.lib.sizes=FALSE]
dim(FtDEG)

##plotting distribution of counts 2
# plotDensities(
#   log(FtDEG$counts),
#   # col = c("red", "blue", "green", "yellow",
#   #         "pink", "orange", "black", "gray"),
#   legend = F,
#   main = "Density of reads after filtering")

#mean lib.size
lib.size <- as.character(
  round(mean(FtDEG$samples$lib.size * 1e-6),
        2))

#plotting library sizes   # INKLUDERT
barplot(
    FtDEG$samples$lib.size*1e-6,
    names=FtDEG$samples$sample,
    ylab="Library size (millions)") +
  abline(
    h=mean(FtDEG$samples$lib.size*1e-6),
    col="Red",
    lty=5, lwd=1) +
  text(
    x=8.5,
    y=20.5,
    paste(
      lib.size,
      "million reads",
      sep = " "),
    col= "black")

#plotting transcripts counts    # IKKE INKLUDERT
# barplot(
#   rowSums(FtDEG$counts*1e-6),
#   las=2,
#   main="Counts per transcript",
#   axisnames = FALSE,
#   ylab = "counts in millions",
#   cex.axis=0.8)
```

```
#normalization
FtDEG <- calcNormFactors(
  FtDEG,
  method = "TMM")

FtDEG$samples

#logcpm values
lcpm <- cpm(
  FtDEG,
  prior.count = 2,
  log=TRUE)

boxplot(
  lcpm,
  xlab= "samples",
  ylab= "log-cpm")

## Variance Stablizing transformation  - VST
vsd <- vst(
  round(FtDEG$counts),
  blind = F) #default blind=T(for totally unsupervised clustering)

## Hierarchical Clustering
sampleTree <- hclust(
  dist(
    t(vsd)))

# plot

plot(
  sampleTree,
  labels = FtDEG$samples$group2,
  main = "Hierarchical Clustering",
  sub = "",
  xlab = "",
  cex.lab = 1.5,
  cex.axis = 1.5,
  cex.main = 2)

# compute pairwise correlations
vsd_cor <- cor(vsd)

#assigning colors
# ann_colors = list(
#   cultivar = c(Engmo = "orange2",
#               Grindstad= "darkgreen"),
#   treatment = c(C = "#084594", T1 = "#6BAED6", T2 =  ))

# run dev.off() if problem
# plotting correlations heatmap

pheatmap::pheatmap(
  vsd_cor,
  clustering_distance_rows = "correlation",
  clustering_distance_cols = "correlation",
  annotation = mdta[,c(3,4)],
  labels_row = mdta$Days,
  labels_col = mdta$Light,
  # annotation_colors = ann_colors,
  angle_col = 45,
  main ="Correlation")



# transpose vst count matrix
vsdt <- t(vsd)

#calculating the variance of the vst transformed counts
var_genes <- apply(vsdt, 2, var)   #kind of a for loop, calculates the variance in all columns

# sorting & picking top 500 highly variable genes for plotting
select_var <- names(
  sort(
    var_genes,
    decreasing=TRUE))[1:500]

# extracting 500 highly variable transcripts from vst matrix
vsdpc <- vsdt[,select_var]   # selecting subset based on the content of select_var

#principle component analysis
pc <- dudi.pca(
  vsdpc,
  center=T,
  scale=F,
  scannf=F,
  nf=6)
summary(pc) #Projected inertia (%) is the variance explained by each pc axis

#visualize the % of variance explained by each principle components
fviz_eig(pc, addlabels = T)

#adding metadata for principle components
pcaData <- as.data.frame(pc$li[,1:6])

# pcaData$group <- FtDEG$samples$group2[match(
#   rownames(pcaData),
#   rownames(FtDEG$samples))]

pcaData$Light <- FtDEG$samples$Light[match(
  rownames(pcaData),
  rownames(FtDEG$samples))]

pcaData$Days <- FtDEG$samples$Days[match(
  rownames(pcaData),
  rownames(FtDEG$samples))]

# pcaData$treatment <- FtDEG$samples$treatment[match(
#   rownames(pcaData),
#   rownames(FtDEG$samples))]

#plotting
ggplot(
  pcaData,
  aes(
    x = Axis1,
    y = Axis2,
    color = Light,
    shape = Days)) +
  geom_point(size =6) +
  xlab("PC1: 43.8% variance") +
  ylab("PC2: 17.3% variance") +
  coord_fixed() +
  ggtitle("PCA (top 500 highly variable genes)") +
  theme(text = element_text(size = 20),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        plot.title=element_text(hjust=0.5))

# model matrix
design <- model.matrix(
  ~ 0+group2,
  FtDEG$samples) # using grouped variable (treatment & cultivar)

# design_alt <- model.matrix(
#   ~ 0+Light+Days+Light:Days,
#   FtDEG$samples) # adding 0 removes intercept

colnames(design) <- levels(FtDEG$samples$group2)

# estimating common dispersion
# FtDEG <- estimateGLMCommonDisp(FtDEG, design = design)
# FtDEG <- estimateGLMTrendedDisp(FtDEG, design = design)
# FtDEG <- estimateGLMTagwiseDisp(FtDEG, design = design)

FtDEG <- estimateDisp(FtDEG, design = design)

#biological coefficient of variance
sqrt(FtDEG$common.dispersion)

# = 0.49

# plotBCV(FtDEG, main = "Biological coefficient of variation")

# making contrasts
# my_contrasts<- makeContrasts(
#   MAX_MIN_1 = D1MAX-D1MIN,
#   MAX_MIN_3 = D3MAX-D3MIN,
#   MAX_MIN_9 = D9MAX-D9MIN,
#
#   MIN_9_0 = D9MIN-C0,
#   MED_9_1 = D9MED-D1MED,
#   MAX_9_1 = D9MAX-D1MAX,
#
#   CUT = C0-NC,
#
#   MED_MIN_1 = D1MED-D1MIN,
#   MED_MIN_3 = D3MED-D3MIN,
#   MED_MIN_9 = D9MED-D9MIN,
#   levels = design)

my_contrasts<- makeContrasts(
  MAX_MIN_1 = MAX1-C1,
  MAX_MIN_3 = MAX3-C3,
  MAX_MIN_9 = MAX9-C9,

  MIN_9_0 = C9-C0,
  MED_9_1 = MED9-MED1,
  MAX_9_1 = MAX9-MAX1,

  CUT = C0-UC,

  levels = design)


# glm fit
fit <- glmFit(FtDEG, design)

# MAX1-C1

fit1 <- glmTreat(
  fit,
  contrast = my_contrasts[,1],
  lfc = log2(1.5)) # should be down or up regulated by about 30 %. the ones below threshold are not
important, but not all of the ones above are nececarily important.

summary(
  decideTests(
    fit1,
    p.value = 0.05))

light1 <- topTags(
  fit1,
  n = nrow(FtDEG),
  adjust.method = "fdr",
  p.value = 0.05)$table %>% rownames_to_column("GeneID")

# write.table(
#   light1,
#   file = "light1_outny.txt",
#   col.names = T,
#   row.names = F,
#   quote = F)


# plotMD(fit1, main = "Mean-Difference plot, MAX1 vs C1")
#        # main.cex=2, cex.main=2, cex.lab=1.5, cex.axis=1.5)


# MAX3-C3

fit2 <- glmTreat(
  fit,
  contrast = my_contrasts[,2],
  lfc = log2(1.5))
```

```
summary(
  decideTests(
    fit2,
    p.value = 0.05))

light3 <- topTags(
  fit2,
  n = nrow(FtDEG),
  adjust.method = "fdr",
  p.value = 0.05)$table %>% rownames_to_column("GeneID")

# write.table(
#   light3,
#   file = "light3_outny.txt",
#   col.names = T,
#   row.names = F)

# plotMD(fit2, main = "Mean-Difference plot, MAX3 vs C3")

# MAX9-C9

fit3 <- glmTreat(
  fit,
  contrast = my_contrasts[,3],
  lfc = log2(1.5))

summary(
  decideTests(
    fit3,
    p.value = 0.05))

light9 <- topTags(
  fit3,
  n = nrow(FtDEG),
  adjust.method = "fdr",
  p.value = 0.05)$table %>% rownames_to_column("GeneID")

# write.table(
#   light9,
#   file = "light9_outny.txt",
#   col.names = T ,
#   row.names = F)

# plotMD(fit3, main = "Mean-Difference plot, MAX9 vs C9")

# C9-C0

fit4 <- glmTreat(
  fit,
  contrast = my_contrasts[,4],
  lfc = log2(1.5))

summary(
  decideTests(
    fit4,
    p.value = 0.05))
```

```
timeMIN <- topTags(
  fit4,
  n = nrow(FtDEG),
  adjust.method = "fdr",
  p.value = 0.05)$table %>% rownames_to_column("GeneID")

# write.table(
#   timeMIN,file = "time40_outny.txt",
#   col.names = T,
#   row.names = F)

# plotMD(fit4, main = "Mean-Difference plot, C9 vs C0")

# MED9-MED1

fit5 <- glmTreat(
  fit,
  contrast = my_contrasts[,5],
  lfc = log2(1.5))

summary(
  decideTests(
    fit5,
    p.value = 0.05))

timeMED <- topTags(
  fit5,
  n=nrow(FtDEG),
  adjust.method = "fdr",
  p.value = 0.05)$table %>% rownames_to_column("GeneID")

# write.table(
#   timeMED,file = "time100_outny.txt",
#   col.names = T,
#   row.names = F)

# plotMD(fit5, main = "Mean-Difference plot, MED9 vs MED1")

# MAX9-MAX1

fit6 <- glmTreat(
  fit,
  contrast = my_contrasts[,6],
  lfc = log2(1.5))

summary(
  decideTests(
    fit6,
    p.value = 0.05))

timeMAX <- topTags(
  fit6,
  n=nrow(FtDEG),
  adjust.method = "fdr",
  p.value = 0.05)$table %>% rownames_to_column("GeneID")

# write.table(
#   timeMAX,file = "time250_outny.txt",
#   col.names = T,
#   row.names = F)

# plotMD(fit6, main = "Mean-Difference plot, MAX9 vs MAX1")

# C0-UC

fit7 <- glmTreat(
  fit,
  contrast = my_contrasts[,7],
  lfc = log2(1.5))

summary(
  decideTests(
    fit7,
    p.value = 0.05))

cut_nocut <- topTags(
  fit7,
  n=nrow(FtDEG),
  adjust.method = "fdr",
  p.value = 0.05)$table %>% rownames_to_column("GeneID")

# write.table(
#   cut_nocut,file = "cut_nocut_outny.txt",
#   col.names = T,
#   row.names = F)

# plotMD(fit7, main = "Mean-Difference plot, C0 vs UC")

# MED1-C1

fit8 <- glmTreat(
  fit,
  contrast = my_contrasts[,8],
  lfc = log2(1.5)) # should be down or up regulated by about 30 %. the ones below threshold are not important, but not all of the ones above are nececarily important.

summary(
  decideTests(
    fit8,
    p.value = 0.05))

lightMED1 <- topTags(
  fit8,
  n = nrow(FtDEG),
  adjust.method = "fdr",
  p.value = 0.05)$table %>% rownames_to_column("GeneID")

# write.table(
#   lightMED1,
#   file = "lightMED1_outny.txt",
#   col.names = T,
#   row.names = F,
#   quote = F)
# plotMD(fit8, main = "Mean-Difference plot, MED1 vs C1")

# MED3-C3

fit9 <- glmTreat(
  fit,
  contrast = my_contrasts[,9],
  lfc = log2(1.5)) # should be down or up regulated by about 30 %. the ones below threshold are not
important, but not all of the ones above are nececarily important.

summary(
  decideTests(
    fit9,
    p.value = 0.05))

lightMED3 <- topTags(
  fit9,
  n = nrow(FtDEG),
  adjust.method = "fdr",
  p.value = 0.05)$table %>% rownames_to_column("GeneID")

# write.table(
#   lightMED3,
#   file = "lightMED3_outny.txt",
#   col.names = T,
#   row.names = F,
#   quote = F)

# plotMD(fit9, main = "Mean-Difference plot, MED3 vs C3")

# MED9-C9

fit10 <- glmTreat(
  fit,
  contrast = my_contrasts[,10],
  lfc = log2(1.5)) # should be down or up regulated by about 30 %. the ones below threshold are not
important, but not all of the ones above are nececarily important.

summary(
  decideTests(
    fit10,
    p.value = 0.05))

lightMED9 <- topTags(
  fit10,
  n = nrow(FtDEG),
  adjust.method = "fdr",
  p.value = 0.05)$table %>% rownames_to_column("GeneID")

# write.table(
#   lightMED9,
#   file = "lightMED9_outny.txt",
#   col.names = T,
#   row.names = F,
#   quote = F)

# plotMD(fit10, main = "Mean-Difference plot, MED9 vs C9")
```

```
dev.off()

# title(ylab="log-fold-change",mgp=c(2, 1, 0))

par(mfrow = c(1,3), cex = 0.78, cex.main = 2, cex.lab = 2)
plotMD(fit1, main = "MAX1 vs C1", ylab = '')
title(ylab="log-fold-change", mgp = c(2.4,1,0))
plotMD(fit2, main = "MAX3 vs C3", ylab = '')
plotMD(fit3, main = "MAX9 vs C9", ylab = '')

# par(mfrow = c(1,3), cex = 1, cex.main = 2, cex.lab = 2)
plotMD(fit4, main = "C9 vs C0", ylab = '')
title(ylab="log-fold-change", mgp = c(2.4,1,0))
plotMD(fit5, main = "MED9 vs MED1", ylab = '')
plotMD(fit6, main = "MAX9 vs MAX1", ylab = '')

# par(mfrow = c(1,4), cex = 0.78, cex.main = 2, cex.lab = 2)
plotMD(fit7, main = "C0 vs UC", ylab = '')
title(ylab="log-fold-change", mgp = c(2.4,1,0))
plotMD(fit8, main = "MED1 vs C1", ylab = '')
plotMD(fit10, main = "MED9 vs C9", ylab = '')

# plotMD(fit9, main = "MED3 vs C3", ylab = '')

# venn plot
x= list(
  "MAX9 vs MAX1" = timeMAX$GeneID,
  "MAX1 vs C1" = light1$GeneID,
  "MED9 vs MED1" = timeMED$GeneID)

ggvenn(
  x,
  fill_color = c("#0073C2FF", "#EFC000FF", "#CD534CFF"),
  show_percentage = F,
  fill_alpha = 0.4,
  set_name_size = 6,
  stroke_color = "black",
  stroke_linetype = 0)

# upset plot

upset <-   list(
  "MAX1 vs C1" = light1$GeneID,
  "MAX3 vs C3" = light3$GeneID,
  "MAX9 vs C9" = light9$GeneID,
  "C9 vs C0" = timeMIN$GeneID,
  "MED9 vs MED1" = timeMED$GeneID,
  "MAX9 vs MAX1" = timeMAX$GeneID,
  "C0 vs UC" = cut_nocut$GeneID)

upset(fromList(upset), nsets = 7,
      mainbar.y.label = "DE genes in intersection",
      sets.x.label = "Number of genes \nin the dataset",
      sets.bar.color = c("red", "green", "blue", "orange", "yellow", "lightblue", "purple" ),
      point.size = 3.5,
      line.size = 1.2,
    text.scale = c(2.8,2.8,2.5,2.5,2.5,2.5))

# c(intersection size title,
#   intersection size tick labels,
#   set size title,
#   set size tick labels,
#   set names,
#   numbers above bars)

upset2 <- list(
  "UC" = round(rowMeans(FtDEG[["counts"]][,4:8])),
  "C0" = round(rowMeans(FtDEG[["counts"]][,1:4])),
  "C1" = round(rowMeans(FtDEG[["counts"]][,9:11])),
  "C3" = round(rowMeans(FtDEG[["counts"]][,12:15])),
  "C9" = round(rowMeans(FtDEG[["counts"]][,16:19])),

  "MED1" = round(rowMeans(FtDEG[["counts"]][,20:23])),
  "MED3" = round(rowMeans(FtDEG[["counts"]][,24:27])),
  "MED9" = round(rowMeans(FtDEG[["counts"]][,28:31])),

  "MAX1" = round(rowMeans(FtDEG[["counts"]][,32:35])),
  "MAX3" = round(rowMeans(FtDEG[["counts"]][,36:39])),
  "MAX9" = round(rowMeans(FtDEG[["counts"]][,40:43])))

upset(fromList(upset2), nsets = 11,
    mainbar.y.label = "Genes in intersection",
    sets.x.label = "Number of genes \nin the dataset",
    sets.bar.color = c("red", "green", "blue", "orange", "yellow", "lightblue", "purple", "darkgreen",
                "darkblue", "pink", "aquamarine"),
    point.size = 3.5,
    line.size = 1.2,
    text.scale = c(2.8,2.8,2.5,2.5,2,2.5))
```

R script 1

```
rm(list=ls())

library(stringr)
library(Bios2cor)
library(seqinr)
library(terra)

setwd("C:/Users/simon/Prosjekter/RNA_seq/r_analysis")

allkeys <- read.table("clusters_new.txt", header = F)

outfls = list.files(pattern = "*_outny.txt", full.names = F)
dfs<-lapply(outfls, FUN=read.table, sep=" ", header=T)
names(dfs)<- substr(outfls,1, nchar(outfls)-4)

list2env(dfs,envir=.GlobalEnv)

light1_trins <- allkeys[,1] %in% light1_outny[,1]
```

```
light3_trins <- allkeys[,1] %in% light3_outny[,1]
light9_trins <- allkeys[,1] %in% light9_outny[,1]
time40_trins <- allkeys[,1] %in% time40_outny[,1]
time100_trins <- allkeys[,1] %in% time100_outny[,1]
time250_trins <- allkeys[,1] %in% time250_outny[,1]
cutno_trins <- allkeys[,1] %in% cut_nocut_outny[,1]

light1_fin <- allkeys[light1_trins,2]
light3_fin <- allkeys[light3_trins,2]
light9_fin <- allkeys[light9_trins,2]
time40_fin <- allkeys[time40_trins,2]
time100_fin <- allkeys[time100_trins,2]
time250_fin <- allkeys[time250_trins,2]
cutno_fin <- allkeys[cutno_trins,2]

assembly_dta <- import.fasta(
  "safekelp_trinity_assembly.Trinity.fasta",
  aa.to.upper = TRUE, gap.to.dash = TRUE, log.file = NULL)

light1_write <- assembly_dta[light1_fin]
light3_write <- assembly_dta[light3_fin]
light9_write <- assembly_dta[light9_fin]
time40_write <- assembly_dta[time40_fin]
time100_write <- assembly_dta[time100_fin]
time250_write <- assembly_dta[time250_fin]
cutno_write <- assembly_dta[cutno_fin]

file.create("light1_extracted4.fasta")
file.create("light3_extracted4.fasta")
file.create("light9_extracted4.fasta")
file.create("time40_extracted4.fasta")
file.create("time100_extracted4.fasta")
file.create("time250_extracted4.fasta")
file.create("cutno_extracted4.fasta")

write.fasta(light1_write, names = names(light1_write), open = "a",
        file.out = "light1_extracted4.fasta", nbchar = 80, as.string = FALSE)
write.fasta(light3_write, names = names(light3_write), open = "a",
        file.out = "light3_extracted4.fasta", nbchar = 80, as.string = FALSE)
write.fasta(light9_write, names = names(light9_write), open = "a",
        file.out = "light9_extracted4.fasta", nbchar = 80, as.string = FALSE)
write.fasta(time40_write, names = names(time40_write), open = "a",
        file.out = "time40_extracted4.fasta", nbchar = 80, as.string = FALSE)
write.fasta(time100_write, names = names(time100_write), open = "a",
        file.out = "time100_extracted4.fasta", nbchar = 80, as.string = FALSE)
write.fasta(time250_write, names = names(time250_write), open = "a",
        file.out = "time250_extracted4.fasta", nbchar = 80, as.string = FALSE)
write.fasta(cutno_write, names = names(cutno_write), open = "a",
        file.out = "cutno_extracted4.fasta", nbchar = 80, as.string = FALSE)
```

R script 1

```
# GO CHARTS
```

```
library(tidyverse)
library(tidyr)
library(data.table)
library(readxl)
library(biomaRt)
library(qdapRegex)
library(clusterProfiler)
library(topGO)
library(WGCNA)

library(ggplot2)
library(dplyr)
library(stringi)
library(readr)
library(forcats)
library(stringr)
library(qdapRegex)
library(tibble)

setwd("C:/Users/simon/Prosjekter/RNA_seq/r_analysis")

################################    1    ################################
rm(list=ls())

allkeys <- read.table("clusters_new.txt", header = F)
colnames(allkeys) <- c("GeneID","TrinID")

annotated1 <- read_tsv("./annotated1_3.tsv")
light1 <- read.table("./light1_outny.txt", header = T)

colnames(annotated1) <- c("qseqid", "sseqid", "bitscore", "evalue", "pident",
                          "goids", "count", "terms", "category")

temp1 <- left_join(allkeys,light1,by=c("GeneID"), keep = F)

colnames(temp1)[2] <- "qseqid"

temp2 <- left_join(temp1,annotated1,by=c("qseqid"), keep=F)

str_which(temp2$terms, "peroxidase")
str_which(temp2$terms, "cellular oxidant")

write.table(temp2[c(324819, 167756, 167772, 198757, 198778, 248844, 248851,
                    248857, 324822, 335874, 335899, 335925,
                    335950, 335975, 336000, 336025),],
            file= "MAX1vsC1_print.txt", sep = "\t")

temp2_omit <- temp2 %>% drop_na()
temp2_omit$log10.pval<--log10(as.numeric(temp2_omit$PValue))
temp2_omit$r_log10.pval <- round(temp2_omit$log10.pval, digit = 2)

for (i in 1:length(temp2_omit$terms)) {
  if (isTRUE(str_detect(temp2_omit$terms[i], ","))) {
    temp2_omit$terms[i] <- strsplit(temp2_omit$terms[i], "[,]", fixed = F)[[1]][1]
  } else {
    next
  }
}


anno_means <- temp2_omit %>%
  group_by(terms) %>%
  dplyr::summarize(Mean = mean(log10.pval, na.rm=T)) %>%
  ungroup()

temp3_omit <- left_join(temp2_omit,anno_means, by=c("terms"), keep= F)
temp3_omit$rmeans <- round(temp3_omit$Mean, digit = 2)

anno_grouped <- temp3_omit %>%
  distinct(terms, .keep_all = T) %>%
  group_by(category) %>%
  slice_max(order_by = rmeans, n= 10) %>%
  ungroup()



ggplot(anno_grouped, aes(x = rmeans, y = fct_reorder(terms, rmeans), fill = category)) +
  geom_bar(stat = "identity") +
  facet_wrap( ~ category, scales = "free_y", nrow = 3) +
  geom_text(
    aes(label = rmeans),
    color = "black",
    hjust = -0.1,
    size = 5.5,
    position = position_dodge(0.9)
  ) +
  ggtitle("Distribution of GO terms for MAX1 vs C1") +
  xlab("-log10(p values)") +
  ylab("Top GO terms grouped by category") +
  theme(
    legend.position = "bottom",
    panel.grid = element_blank(),
    panel.background = element_blank(),
    axis.ticks = element_blank(),
    strip.text.x = element_text(size = 20, face = "bold"),
    strip.background = element_blank(),
    text = element_text(size = 19),
    axis.title.x = element_text(size = 20),
    axis.text.x = element_text(size = 14),
    axis.title.y = element_text(size = 20))

###############################    3    ###############################
rm(list=ls())

allkeys <- read.table("clusters_new.txt", header = F)
colnames(allkeys) <- c("GeneID","TrinID")

annotated1 <- read_tsv("./annotated3_3.tsv")
light1 <- read.table("./light3_outny.txt", header = T)

colnames(annotated1) <- c("qseqid", "sseqid", "bitscore", "evalue",
                          "pident", "goids", "count", "terms", "category")
```

```
temp1 <- left_join(allkeys,light1,by=c("GeneID"), keep = F)

colnames(temp1)[2] <- "qseqid"

temp2 <- left_join(temp1,annotated1,by=c("qseqid"), keep=F)

str_which(temp2$terms, "peroxidase")
str_which(temp2$terms, "cellular oxidant")

write.table(temp2[c(248083, 248092, 240454, 240458,
                    240460, 248086, 248095),], file = "MAX3vsC3_print.txt",
            sep = "\t")

temp2_omit <- temp2 %>% drop_na()
temp2_omit$log10.pval<--log10(as.numeric(temp2_omit$PValue))
temp2_omit$r_log10.pval <- round(temp2_omit$log10.pval, digit = 2)


for (i in 1:length(temp2_omit$terms)) {
  if (isTRUE(str_detect(temp2_omit$terms[i], ","))) {
    temp2_omit$terms[i] <- strsplit(temp2_omit$terms[i], "[,]", fixed = F)[[1]][1]
  } else {
    next
  }
}



anno_means <- temp2_omit %>%
  group_by(terms) %>%
  dplyr::summarize(Mean = mean(log10.pval, na.rm=T)) %>%
  ungroup()

temp3_omit <- left_join(temp2_omit,anno_means, by=c("terms"), keep= F)
temp3_omit$rmeans <- round(temp3_omit$Mean, digit = 2)

anno_grouped <- temp3_omit %>%
  distinct(terms, .keep_all = T) %>%
  group_by(category) %>% # group your data based on the variable Rating
  slice_max(order_by = rmeans, n= 10) %>%
  ungroup()

ggplot(anno_grouped, aes(x = rmeans, y = fct_reorder(terms, rmeans), fill = category)) +
  geom_bar(stat = "identity") +
  facet_wrap( ~ category, scales = "free_y", nrow = 3) +
  geom_text(
    aes(label = rmeans),
    color = "black",
    hjust = -0.1,
    size = 5.5,
    position = position_dodge(0.9)) +
  ggtitle("Distribution of GO terms for MAX3 vs C3") +
  xlab("-log10(p values)") +
  ylab("Top GO terms grouped by category") +
  theme(
    legend.position = "bottom",
    panel.grid = element_blank(),
    panel.background = element_blank(),
    axis.ticks = element_blank(),
    strip.text.x = element_text(size = 20, face = "bold"),
    strip.background = element_blank(),
    text = element_text(size = 19),
    axis.title.x = element_text(size = 20),
    axis.text.x = element_text(size = 14),
    axis.title.y = element_text(size = 20))

################################    9    ################################
rm(list=ls())

allkeys <- read.table("clusters_new.txt", header = F)
colnames(allkeys) <- c("GeneID","TrinID")

annotated1 <- read_tsv("./annotated9_3.tsv")
light1 <- read.table("./light9_outny.txt", header = T)

colnames(annotated1) <- c("qseqid", "sseqid", "bitscore", "evalue",
                          "pident", "goids", "count", "terms", "category")

temp1 <- left_join(allkeys,light1,by=c("GeneID"), keep = F)

colnames(temp1)[2] <- "qseqid"

temp2 <- left_join(temp1,annotated1,by=c("qseqid"), keep=F)

str_which(temp2$terms, "peroxidase")
str_which(temp2$terms, "cellular oxidant")

write.table(temp2[c(300057, 240387, 240390, 240392, 300058),],
            file = "MAX9vsC9_print.txt", sep = "\t")

temp2_omit <- temp2 %>% drop_na()
temp2_omit$log10.pval<--log10(as.numeric(temp2_omit$PValue))
temp2_omit$r_log10.pval <- round(temp2_omit$log10.pval, digit = 2)



for (i in 1:length(temp2_omit$terms)) {
  if (isTRUE(str_detect(temp2_omit$terms[i], ","))) {
    temp2_omit$terms[i] <- strsplit(temp2_omit$terms[i], "[,]", fixed = F)[[1]][1]
  } else {
    next
  }
}



anno_means <- temp2_omit %>%
  group_by(terms) %>%
  dplyr::summarize(Mean = mean(log10.pval, na.rm=T)) %>%
  ungroup()

temp3_omit <- left_join(temp2_omit,anno_means, by=c("terms"), keep= F)
temp3_omit$rmeans <- round(temp3_omit$Mean, digit = 2)
```

```
anno_grouped <- temp3_omit %>%
  distinct(terms, .keep_all = T) %>%
  group_by(category) %>% # group your data based on the variable Rating
  slice_max(order_by = rmeans, n= 10) %>%
  ungroup()

ggplot(anno_grouped, aes(x = rmeans, y = fct_reorder(terms, rmeans), fill = category)) +
  geom_bar(stat = "identity") +
  facet_wrap( ~ category, scales = "free_y", nrow = 3) +
  geom_text(
    aes(label = rmeans),
    color = "black",
    hjust = -0.1,
    size = 5.5,
    position = position_dodge(0.9)
  ) +
  ggtitle("Distribution of GO terms for MAX9 vs C9") +
  xlab("-log10(p values)") +
  ylab("Top GO terms grouped by category") +
  theme(
    legend.position = "bottom",
    panel.grid = element_blank(),
    panel.background = element_blank(),
    axis.ticks = element_blank(),
    strip.text.x = element_text(size = 20, face = "bold"),
    strip.background = element_blank(),
    text = element_text(size = 19),
    axis.title.x = element_text(size = 20),
    axis.text.x = element_text(size = 14),
    axis.title.y = element_text(size = 20))
```

R script 4

```
rm(list= ls())

library(tidyverse)
library(tidyr)
library(readxl)
library(qdapRegex)
library(clusterProfiler)
library(topGO)
library(ggplot2)
library(dplyr)
library(forcats)
library(stringr)
library(readr)
library(forcats)
library(stringr)
library(edgeR)
library(stats)
library(gplots)
library(tibble)
library(RColorBrewer)
library(BBmisc)

## import
setwd("C:/Users/simon/Prosjekter/RNA_seq/r_analysis")

## counts
counts <- read.table("counts.txt", header = T)
counts <- rownames_to_column(counts,var = "GeneID")
counts <- as_tibble(counts)

counts$C0 <- rowMeans(counts[,2:5])
counts$UC <- rowMeans(counts[,6:9])
counts$C1 <- rowMeans(counts[,10:12])
counts$C3 <- rowMeans(counts[,13:16])
counts$C9 <- rowMeans(counts[,17:20])
counts$MED1 <- rowMeans(counts[,21:24])
counts$MED3 <- rowMeans(counts[,25:28])
counts$MED9 <- rowMeans(counts[,29:32])
counts$MAX1 <- rowMeans(counts[,33:36])
counts$MAX3 <- rowMeans(counts[,37:40])
counts$MAX9 <- rowMeans(counts[,41:44])

counts <- counts %>%
  select(-c(2:44))

## contrast files
fls <- list.files(pattern = "*_outny.txt", full.names = F)
dfs <- lapply(fls, FUN=read.table, sep="", header=T)
dfs <- lapply(dfs, FUN=as_tibble, sep="", header=T)
dfs <- lapply(dfs[], FUN=select, -c(3,5:6))
names(dfs)<- substr(fls,1, nchar(fls)-10)
list2env(dfs,envir=.GlobalEnv)
colnames(light1) <- c('GeneID', 'logFC_1', 'logCPM_1')
colnames(light3) <- c('GeneID', 'logFC_3', 'logCPM_3')
colnames(light9) <- c('GeneID', 'logFC_9', 'logCPM_9')
colnames(time40) <- c('GeneID', 'logFC_40', 'logCPM_40')
colnames(time100) <- c('GeneID', 'logFC_100', 'logCPM_100')
colnames(time250) <- c('GeneID', 'logFC_250', 'logCPM_250')
colnames(cut_nocut) <- c('GeneID', 'logFC_cut', 'logCPM_cut')

contrasts <- full_join(light1, light3, by = c("GeneID"), keep = F)
contrasts <- full_join(contrasts, light9, by = c('GeneID'), keep = F)
contrasts <- full_join(contrasts, time40, by = c('GeneID'), keep = F)
contrasts <- full_join(contrasts, time100, by = c('GeneID'), keep = F)
contrasts <- full_join(contrasts, time250, by = c('GeneID'), keep = F)
contrasts <- full_join(contrasts, cut_nocut, by = c('GeneID'), keep = F)

rm(list= c('light1','light3','light9','time40','time100','time250','cut_nocut'))

## keys
keys <- read.table("clusters_new.txt", header = F)
colnames(keys) <- c("GeneID","TrinID")

## blast results
afls <- list.files(pattern = "*_3.tsv", full.names = F)
adfs <- lapply(afls, FUN=read.table, sep="\t", header=T, na.strings = '-')
adfs <- lapply(adfs, FUN=as_tibble, sep="", header=T)
adfs <- lapply(adfs[], FUN=select, -c(3:9))
adfs <- lapply(adfs[], FUN=distinct, .keep_all = T)
names(adfs)<- substr(afls,1, nchar(afls)-6)
list2env(adfs,envir=.GlobalEnv)
colnames(annotated1) <- c('TrinID', 'sseqid')
colnames(annotated3) <- c('TrinID', 'sseqid')
colnames(annotated9) <- c('TrinID', 'sseqid')
colnames(annotated40) <- c('TrinID', 'sseqid')
colnames(annotated100) <- c('TrinID', 'sseqid')
colnames(annotated250) <- c('TrinID', 'sseqid')
colnames(annotatedcutno) <- c('TrinID', 'sseqid')

annotations <- full_join(annotated1, annotated3, by = c("TrinID", 'sseqid'), keep = F)
annotations <- full_join(annotations, annotated9, by = c('TrinID', 'sseqid'), keep = F)
annotations <- full_join(annotations, annotated40, by = c('TrinID', 'sseqid'), keep = F)
annotations <- full_join(annotations, annotated100, by = c('TrinID', 'sseqid'), keep = F)
annotations <- full_join(annotations, annotated250, by = c('TrinID', 'sseqid'), keep = F)
annotations <- full_join(annotations, annotatedcutno, by = c('TrinID', 'sseqid'), keep = F)

rm(list=
c('annotated1','annotated3','annotated9','annotated40','annotated100','annotated250','annotatedcutno'))

annotations <- annotations %>%
  drop_na(sseqid)

annotations <- filter(annotations, !(str_detect(annotations$sseqid[], 'Uncharacterized')))

for (i in 1:length(annotations$sseqid)) {
  if (str_detect(annotations$sseqid[i], 'Uncharacterized')) {
    next
  } else {
    annotations$sseqid[i] <- strsplit(annotations$sseqid[i], "[_]", fixed = F)[[1]][2]
    annotations$sseqid[i] <- strsplit(annotations$sseqid[i], "OS", fixed = F)[[1]][1]
  }
}

# annogrep <- annotations %>%
#   filter(grepl('peroxidase|cellular oxidant', sseqid))

# annogrep <- left_join(annogrep, keys, by = "TrinID")
# annogrep <- left_join(annogrep, contrasts, by = "GeneID")

annotations <- inner_join(annotations, keys, by = c('TrinID'), keep = F)

annotations <- annotations %>%
  distinct(GeneID, .keep_all = T)


################ annotation and counts ################




anno_counts <- inner_join(annotations, counts, by = c('GeneID'), keep = F)

anno_medians0 <- anno_counts %>%
  group_by(sseqid) %>%
  dplyr::summarize(C0 = max(C0, na.rm=T)) %>%
  ungroup()
anno_medians1 <- anno_counts %>%
  group_by(sseqid) %>%
  dplyr::summarize(UC = max(UC, na.rm=T)) %>%
  ungroup()
anno_medians2 <- anno_counts %>%
  group_by(sseqid) %>%
  dplyr::summarize(C1 = max(C1, na.rm=T)) %>%
  ungroup()
anno_medians3 <- anno_counts %>%
  group_by(sseqid) %>%
  dplyr::summarize(C3 = max(C3, na.rm=T)) %>%
  ungroup()
anno_medians4 <- anno_counts %>%
  group_by(sseqid) %>%
  dplyr::summarize(C9 = max(C9, na.rm=T)) %>%
  ungroup()
anno_medians5 <- anno_counts %>%
  group_by(sseqid) %>%
  dplyr::summarize(MED1 = max(MED1, na.rm=T)) %>%
  ungroup()
anno_medians6 <- anno_counts %>%
  group_by(sseqid) %>%
  dplyr::summarize(MED3 = max(MED3, na.rm=T)) %>%
  ungroup()
anno_medians7 <- anno_counts %>%
  group_by(sseqid) %>%
  dplyr::summarize(MED9 = max(MED9, na.rm=T)) %>%
  ungroup()
anno_medians8 <- anno_counts %>%
  group_by(sseqid) %>%
  dplyr::summarize(MAX1 = max(MAX1, na.rm=T)) %>%
  ungroup()
anno_medians9 <- anno_counts %>%
  group_by(sseqid) %>%
  dplyr::summarize(MAX3 = max(MAX3, na.rm=T)) %>%
  ungroup()
anno_medians10 <- anno_counts %>%
  group_by(sseqid) %>%
  dplyr::summarize(MAX9 = max(MAX9, na.rm=T)) %>%
  ungroup()

medians <- full_join(anno_medians0, anno_medians1, by = c('sseqid'), keep = F)
medians <- full_join(medians, anno_medians2, by = c('sseqid'), keep = F)
medians <- full_join(medians, anno_medians3, by = c( 'sseqid'), keep = F)
medians <- full_join(medians, anno_medians4, by = c( 'sseqid'), keep = F)
medians <- full_join(medians, anno_medians5, by = c( 'sseqid'), keep = F)
medians <- full_join(medians, anno_medians6, by = c( 'sseqid'), keep = F)
medians <- full_join(medians, anno_medians7, by = c( 'sseqid'), keep = F)
medians <- full_join(medians, anno_medians8, by = c( 'sseqid'), keep = F)
medians <- full_join(medians, anno_medians9, by = c( 'sseqid'), keep = F)
medians <- full_join(medians, anno_medians10, by = c( 'sseqid'), keep = F)


rm(list=c('anno_medians0','anno_medians1','anno_medians2','anno_medians3','anno_medians4',
          'anno_medians5','anno_medians6','anno_medians7','anno_medians8','anno_medians9',
        'anno_medians10'))

catmat <- medians[,-c(3,7:9)]



################ annotation and contrasts ################



# anno_contrasts <- inner_join(annotations, contrasts, by = c('GeneID'), keep = F)
#
# anno_medians0 <- anno_contrasts %>%
#   group_by(sseqid) %>%
#   dplyr::summarize(MAX1_C1_FC = max(logFC_1, na.rm=T)) %>%
#   ungroup()
# anno_medians1 <- anno_contrasts %>%
#   group_by(sseqid) %>%
#   dplyr::summarize(MAX3_C3_FC = max(logFC_3, na.rm=T)) %>%
#   ungroup()
# anno_medians2 <- anno_contrasts %>%
#   group_by(sseqid) %>%
#   dplyr::summarize(MAX9_C9_FC = max(logFC_9, na.rm=T)) %>%
#   ungroup()
# anno_medians3 <- anno_contrasts %>%
#   group_by(sseqid) %>%
#   dplyr::summarize(C9_C0_FC = max(logFC_40, na.rm=T)) %>%
#   ungroup()
# anno_medians4 <- anno_contrasts %>%
#   group_by(sseqid) %>%
#   dplyr::summarize(MED9_MED1_FC = max(logFC_100, na.rm=T)) %>%
#   ungroup()
# anno_medians5 <- anno_contrasts %>%
#   group_by(sseqid) %>%
#   dplyr::summarize(MAX9_MAX1_FC = max(logFC_250, na.rm=T)) %>%
#   ungroup()
# anno_medians6 <- anno_contrasts %>%
#   group_by(sseqid) %>%
#   dplyr::summarize(C0_UC_FC = max(logFC_cut, na.rm=T)) %>%
#   ungroup()
#
# anno_medians7 <- anno_contrasts %>%
#   group_by(sseqid) %>%
#   dplyr::summarize(MAX1_C1_CPM = max(logCPM_1, na.rm=T)) %>%
#   ungroup()
# anno_medians8 <- anno_contrasts %>%
#   group_by(sseqid) %>%
#   dplyr::summarize(MAX3_C3_CPM = max(logCPM_3, na.rm=T)) %>%
#   ungroup()
# anno_medians9 <- anno_contrasts %>%
#   group_by(sseqid) %>%
#   dplyr::summarize(MAX9_C9_CPM = max(logCPM_9, na.rm=T)) %>%
#   ungroup()
# anno_medians10 <- anno_contrasts %>%
#   group_by(sseqid) %>%
#   dplyr::summarize(C9_C0_CPM = max(logCPM_40, na.rm=T)) %>%
#   ungroup()
# anno_medians11 <- anno_contrasts %>%
#   group_by(sseqid) %>%
#   dplyr::summarize(MED9_MED1_CPM = max(logCPM_100, na.rm=T)) %>%
#   ungroup()
# anno_medians12 <- anno_contrasts %>%
#   group_by(sseqid) %>%
#   dplyr::summarize(MAX9_MAX1_CPM = max(logCPM_250, na.rm=T)) %>%
#   ungroup()
# anno_medians13 <- anno_contrasts %>%
#   group_by(sseqid) %>%
#   dplyr::summarize(C0_UC_CPM = max(logCPM_cut, na.rm=T)) %>%
#   ungroup()
#
# medians2 <- full_join(anno_medians0, anno_medians1, by = c('sseqid'), keep = F)
# medians2 <- full_join(medians2, anno_medians2, by = c('sseqid'), keep = F)
# medians2 <- full_join(medians2, anno_medians3, by = c( 'sseqid'), keep = F)
# medians2 <- full_join(medians2, anno_medians4, by = c( 'sseqid'), keep = F)
# medians2 <- full_join(medians2, anno_medians5, by = c( 'sseqid'), keep = F)
# medians2 <- full_join(medians2, anno_medians6, by = c( 'sseqid'), keep = F)
# medians2 <- full_join(medians2, anno_medians7, by = c( 'sseqid'), keep = F)
# medians2 <- full_join(medians2, anno_medians8, by = c( 'sseqid'), keep = F)
# medians2 <- full_join(medians2, anno_medians9, by = c( 'sseqid'), keep = F)
# medians2 <- full_join(medians2, anno_medians10, by = c( 'sseqid'), keep = F)
# medians2 <- full_join(medians2, anno_medians11, by = c( 'sseqid'), keep = F)
# medians2 <- full_join(medians2, anno_medians12, by = c( 'sseqid'), keep = F)
# medians2 <- full_join(medians2, anno_medians13, by = c( 'sseqid'), keep = F)
#
# rm(list=c('anno_medians0','anno_medians1','anno_medians2','anno_medians3','anno_medians4',
#           'anno_medians5','anno_medians6','anno_medians7','anno_medians8','anno_medians9',
#           'anno_medians10','anno_medians11','anno_medians12','anno_medians13'))
#
# medians2sel <- medians2[,c(1,2,6,7,9,13,14)]
#
#
# ## catmat ##
#
# catmat <- inner_join(medians1sel, medians2sel, by = c('sseqid'), keep = F)



mypalette <- brewer.pal(11,"RdYlBu")
morecols <- colorRampPalette(mypalette)
# col.cell <- c("purple","orange")[sampleinfo$CellType]


################ plotting ################

catmat1 <- column_to_rownames(catmat, var = "sseqid")
catmatcpm <- cpm(catmat1[1:7], log = T)
var_genes <- apply(catmatcpm, 1, var)

select_var <- names(sort(var_genes, decreasing=TRUE))[1:10]

highly_variable_cpm <- catmatcpm[select_var,]

#write.table(highly_variable_cpm, file = "cpm1_ut.tsv", sep = '\t')
```

```
highly_variable_cpm <- read_xlsx("cpm1_inn.xlsx")

highly_variable_cpm <- column_to_rownames(highly_variable_cpm, var = "...1")

highly_variable_cpm <- as.matrix(highly_variable_cpm)

normcpm <- normalize(
  highly_variable_cpm,
  method = "range",
  range = c(0,1),
  margin = 1,
  on.constant = "quiet")

pheatmap::pheatmap(
  normcpm,
  #scale = "column",
  # clustering_distance_rows = "correlation",
  # clustering_distance_cols = "correlation",
  # cluster_rows = T,
  # cluster_cols = T,
  # annotation = mdta[,c(4,5)],
  # labels_row = mdta$Days,
  # labels_col = mdta$Light,
  annotation_colors = rev(morecols(50)),
  angle_col = 45,
  fontsize = 20,
  main ="Top 10 most variable genes across samples")

################ chosen genes ################

catgrep <- catmat %>%
  filter(grepl('vanadium|Vanadium|xanthin|glutathione|Glutathione', sseqid))

test1 <- anno_counts %>%
  distinct(sseqid, .keep_all = T)

test2 <- inner_join(catgrep, test1, by = "sseqid", keep = F)

test3 <- test2[,c(1,10)]

test4 <- inner_join(test3, contrasts, by = "GeneID")

catcpm2 <- column_to_rownames(catgrep, var = "sseqid")

#write.table(catcpm2, file = "catgrep_ut3.tsv", sep = '\t')

catgrep_inn <- read_xlsx("catgrep_inn3.xlsx")

catgrep_inn <- column_to_rownames(catgrep_inn, var = "...1")

normcat <- cpm(catgrep_inn[1:7], log = T)

normcat <- normalize(
  catgrep_inn,
  method = "standardize",
  range = c(0,1),
  margin = 1,
  on.constant = "quiet"
)
colnames(normcat) <- colnames(catgrep_inn)


highly_variable_cpm <- as.matrix(normcat)

pheatmap::pheatmap(
  highly_variable_cpm,
  scale = "row",
  #clustering_distance_rows = "correlation",
  # clustering_method = "complete",
  # clustering_distance_cols = "correlation",
  #cluster_rows = T,
  # cluster_cols = T,
  # annotation = mdta[,c(4,5)],
  # labels_row = mdta$Days,
  # labels_col = mdta$Light,
  annotation_colors = rev(morecols(50)),
  angle_col = 45,
  fontsize = 20,
  main ="Genes associated with vHPOs and heavy metal")
```

SSH scripts

SSH script 1

```
## TRINITY

module load Anaconda3/5.3.0

source activate Trinity

cd /mnt/SCRATCH/simonha/link_trinity2

# trinity denovo assembly

Trinity --seqType fq --max_memory 110G --CPU 110 \
	--left forward_1.fq.gz --right reverse_2.fq.gz \
	--min_contig_length 200 --SS_lib_type FR --full_cleanup \
	--min_kmer_cov 2 --output /mnt/SCRATCH/simonha/safekelp_trinity_assembly

source deactivate

## KRAKEN2

cd /mnt/SCRATCH/simonha/Kraken4

#to decontaminate a denovo assembly / single end reads

kraken2 -db /mnt/SCRATCH/simonha/Kraken2/db_deconta
/mnt/SCRATCH/simonha/safekelp_trinity_assembly.Trinity.fasta \
--threads 70 --classified-out /mnt/SCRATCH/simonha/Kraken4/class/cseqs.fasta --unclassified-out /mnt/SCRATCH/simonha/Kraken4/unclass/unseqs.fasta \
--output . --report ./report_Kraken4

## FASTQC

module load Anaconda3/5.3.0

source activate Quality

cd /mnt/users/simonha/RNA_SEQ/link

# fastqc

fastqc *.fq.gz

source deactivate


## MULTIQC

module load Anaconda3/5.3.0

source activate Quality

cd /mnt/users/simonha/RNA_SEQ/link

multiqc * -o .

source deactivate

## BUSCO

conda activate /mnt/SCRATCH2/IPV-RNAseq22/anaconda3/envs/Busco

busco -m transcriptome -i ../safekelp_trinity_assembly.Trinity.fasta -o BUSCO_out -l embryophyta_odb10 -f -c 20

conda deactivate

## ABUNDANCE ESTIMATION

cd /mnt/SCRATCH/simonha/alignment

# use find . -type f -name "*.isoforms.results" to find paths to all files

abundance_estimates_to_matrix.pl --est_method RSEM --out_prefix FT --cross_sample_norm TMM --name_sample_by_basedir \
--gene_trans_map /mnt/SCRATCH/simonha/safekelp_trinity_assembly.Trinity.fasta.gene_trans_map \
./K1/RSEM.isoforms.results \
./K2/RSEM.isoforms.results \
./K3/RSEM.isoforms.results \
./K4/RSEM.isoforms.results \
./K5/RSEM.isoforms.results \
./K6/RSEM.isoforms.results \
./K7/RSEM.isoforms.results \
./K8/RSEM.isoforms.results \
./K10/RSEM.isoforms.results \
./K11/RSEM.isoforms.results \
./K12/RSEM.isoforms.results \
./K13/RSEM.isoforms.results \
./K14/RSEM.isoforms.results \
./K15/RSEM.isoforms.results \
./K16/RSEM.isoforms.results \
./K17/RSEM.isoforms.results \
./K18/RSEM.isoforms.results \
./K19/RSEM.isoforms.results \
./K20/RSEM.isoforms.results \
./K21/RSEM.isoforms.results \
./K22/RSEM.isoforms.results \
./K23/RSEM.isoforms.results \
./K24/RSEM.isoforms.results \
./K25/RSEM.isoforms.results \
./K26/RSEM.isoforms.results \
./K27/RSEM.isoforms.results \
./K28/RSEM.isoforms.results \
./K29/RSEM.isoforms.results \
./K30/RSEM.isoforms.results \
./K31/RSEM.isoforms.results \
./K32/RSEM.isoforms.results \
./K33/RSEM.isoforms.results \
./K34/RSEM.isoforms.results \
./K35/RSEM.isoforms.results \
./K36/RSEM.isoforms.results \
./K37/RSEM.isoforms.results \
./K38/RSEM.isoforms.results \
./K39/RSEM.isoforms.results \
./K40/RSEM.isoforms.results \
./K41/RSEM.isoforms.results \
./K42/RSEM.isoforms.results \
./K43/RSEM.isoforms.results \
./K44/RSEM.isoforms.results \

## ALIGNMENT

cd /mnt/SCRATCH/simonha/alignment

#quantification
for infile in K1_1.clean.fq.gz K2_1.clean.fq.gz K3_1.clean.fq.gz K4_1.clean.fq.gz K5_1.clean.fq.gz
K6_1.clean.fq.gz K7_1.clean.fq.gz K8_1.clean.fq.gz K10_1.clean.fq.gz K11_1.clean.fq.gz
K12_1.clean.fq.gz K13_1.clean.fq.gz K14_1.clean.fq.gz K15_1.clean.fq.gz K16_1.clean.fq.gz
K17_1.clean.fq.gz K18_1.clean.fq.gz K19_1.clean.fq.gz K20_1.clean.fq.gz K21_1.clean.fq.gz
K22_1.clean.fq.gz K23_1.clean.fq.gz K24_1.clean.fq.gz K25_1.clean.fq.gz K26_1.clean.fq.gz
K27_1.clean.fq.gz K28_1.clean.fq.gz K29_1.clean.fq.gz K30_1.clean.fq.gz K31_1.clean.fq.gz
K32_1.clean.fq.gz K33_1.clean.fq.gz K34_1.clean.fq.gz K35_1.clean.fq.gz K36_1.clean.fq.gz
K37_1.clean.fq.gz K38_1.clean.fq.gz K39_1.clean.fq.gz K40_1.clean.fq.gz K41_1.clean.fq.gz
K42_1.clean.fq.gz K43_1.clean.fq.gz K44_1.clean.fq.gz;
do
base=$(basename ${infile} _1.clean.fq.gz)
align_and_estimate_abundance.pl --seqType fq --SS_lib_type FR \
                              --transcripts /mnt/SCRATCH/simonha/alignment/unseqs.fasta \
                              --left <(zcat ${infile}) --right <(zcat ${base}_2.clean.fq.gz) --est_method RSEM --
aln_method bowtie2 \
                    --trinity_mode --prep_reference --output_dir /mnt/SCRATCH/simonha/alignment/${base} --thread_count 80 2>&1 > ${base}_opt
done

#S1_1.trimm.fq.gz  S2_1.trimm.fq.gz  S49_1.trimm.fq.gz  S38_1.trimm.fq.gz S43_1.trimm.fq.gz S44_1.trimm.fq.gz  S7_1.trimm.fq.gz  S8_1.trimm.fq.gz

## CORSET

cd /mnt/SCRATCH/simonha/corset

####corset command###


###
# /mnt/users/simonha/software/corset-1.09-linux64/corset *.bam -r true-stop

for FILE in `ls *.bam` ; do
  /mnt/users/simonha/software/corset-1.09-linux64/corset -r true-stop $FILE &
done
wait

## CORSET 2

cd /mnt/SCRATCH/simonha/corset

####corset command###

####

corset K1.bam.corset-reads K2.bam.corset-reads K3.bam.corset-reads K4.bam.corset-reads K5.bam.corset-reads K6.bam.corset-reads \
K7.bam.corset-reads K8.bam.corset-reads K10.bam.corset-reads K11.bam.corset-reads K12.bam.corset-reads K13.bam.corset-reads \
K14.bam.corset-reads K15.bam.corset-reads K16.bam.corset-reads K17.bam.corset-reads K18.bam.corset-reads K19.bam.corset-reads \
K20.bam.corset-reads K21.bam.corset-reads K22.bam.corset-reads K23.bam.corset-reads K24.bam.corset-reads K25.bam.corset-reads \
K26.bam.corset-reads K27.bam.corset-reads K28.bam.corset-reads K29.bam.corset-reads K30.bam.corset-reads K31.bam.corset-reads \
K32.bam.corset-reads K33.bam.corset-reads K34.bam.corset-reads K35.bam.corset-reads K36.bam.corset-reads K37.bam.corset-reads \
K38.bam.corset-reads K39.bam.corset-reads K40.bam.corset-reads K41.bam.corset-reads K42.bam.corset-reads K43.bam.corset-reads \
K44.bam.corset-reads \
 -n K1,K2,K3,K4,K5,K6,K7,K8,K10,K11,K12,K13,K14,K15,K16,K17,K18,K19,K20,K21,K22,K23,K24,K25,K26,K27,K28,K29,K30,K31,K32,K33,K34,K35,K36,K37,K38,K39,K40,K41,K42,K43,K44 \
 -g 1,1,1,1,2,2,2,2,3,3,3,3,4,4,4,4,5,5,5,5,6,6,6,6,7,7,7,7,8,8,8,8,9,9,9,9,10,10,10,10,11,11,11,11 \
 -d 0.3 -m 10 -i corset -f true \


## DIAMOND

diamond blastx --db ./phaeophyceae_db.dmnd -q ./light1_extracted.fasta -o matches1.xml --outfmt 5 #--ultra-sensitive
```

Norges miljø- og biovitenskapelige universitet
Noregs miljø- og biovitskapelege universitet
Norwegian University of Life Sciences

Postboks 5003
NO-1432 Ås
Norway