Title: First *de novo* transcriptome analysis of the Antarctic springtail *Cryptopygus terranovus* (*Collembola: Isotomidae*) following long term heat exposure

Authors: CLAUDIO CUCINI[1], CHIARA LEO[2], FRANCESCO NARDI[1], SAMUELE GRECO[3], CHIARA MANFRIN[3], PIERO G. GIULIANINI[3] and ANTONIO CARAPELLI[1]

Affiliations:

[1] *Department of Life Sciences, University of Siena, 53100 Siena, Italy*

[2] *Department of Life Sciences, Imperial College London, SW7 2AZ London, UK*

[3]*Department of Life Sciences, University of Trieste, 34127 Trieste, Italy*

Correspondance to: claudio.cucini@student.unisi.it (C. Cucini)

Supplemental Material

Supplemental Protocol S1. Differential expression analysis of transcripts performed in the R environment v3.6.3 with the NOISeq package (Taranzona *et al*. 2015)

```
library('DESeq2')
library('NOISeq')
library('ggplot2')
library('RColorBrewer')
library('edgeR')
```

```
library('gplots')


#importing the transcript annotation table previously obtained with a custom python script

annotation=data.frame((read.csv('/home/CC/Cryptopygus_transcript_annotation.tsv', sep= '\t', header= T)))

#importing the gene expression table previously obtained with SALMON

TPM= data.matrix(read.table("/home/CC/Cryptopygus_gene_expression.tsv", header = T, row.names = 1))

#defining the conditions and the sample names

sampleConditions = c("control", "control", "control", "heated", "heated", "heated")

names = c("Ha", "Hb", "Hc", "SSC1a", "SSC1b", "SSC1c")

myFactors= data.frame(condition = sampleConditions, name = names)

#including the TPM expression and the factors in a unique object

myData <- readData(data = TPM , factors = myFactors)


#PCA plot

PCA <- dat(myData, type="PCA")

explo.plot(PCA, factor='condition', plottype='scores')


#DEGs obtained through the noiseqbio function

result <- noiseqbio(myData, norm = c('n'), filter = 1, factor = "condition")

head(result@results)

#filter on the probability treshold and Log 2 fold change

result.deg = degenes(result, q= 0.9999, M  = NULL)

result.deg=result.deg[abs(result.deg$log2FC)>2,]

#filter on the probability treshold and Log 2 fold change of up-regulated genes (=control samples)

result.ct = degenes(result, q= 0.9999, M  = 'UP')

result.ct=result.ct[abs(result.ct$log2FC)>2,]

#filter on the probability treshold and Log 2 fold change of down-regulated genes (=treated samples)

result.he = degenes(result, q= 0.9999, M  = 'DOWN')
```

```
result.he=result.he[abs(result.he$log2FC)>2,]


#heatmap plot

coul <- colorRampPalette(c('black','white', 'red'))

heatmap.2(as.matrix(log10(cpm(TPM)+1)[rownames(result.deg),]),
scale="column", col=coul,trace = 'none')


#writing the results as a table outputs

write.csv(result.deg, '/home/CC/DEGs.csv')

write.csv(result.ct, '/home/CC/DEGs_CT.csv')

write.csv(result.he, '/home/CC/DEGs_HE.csv')
```

Supplemental Protocol S2. GO enrichment analysis performed with python3.7

```
import pandas as pd

import numpy as np

from scipy.stats import hypergeom

import matplotlib.pyplot as plt




#importing the transcript annotation table previously obtained with a custom python script

annotation = pd.read_csv("/home/CC/Cryptopygus_transcript_annotation.tsv", sep = "\t")

#importing the DEGs table previously obtained with the NOISeq package

degs = pd.read_csv("/home/CC/DEGs.csv", sep= ",")

degs.rename(columns={'Unnamed: 0':'transcript_id'}, inplace=True)

#importing the gene expression table previously obtained with SALMON

tpm = pd.read_csv("/home/CC/Cryptopygus_gene_expression.tsv", sep = "\t")

tpm.index = tpm.Name

tpm = tpm[["Ha", "Hb", "Hc", "SSC1a", "SSC1b", "SSC1c", ]]




#merging the annotation dataframe with the DEGs

df = degs.merge(annotation, on = "transcript_id", how = "left")

#filtering the resulting dataframe by a cutoff probability and their log2fold-change

df_filtered = df[(df.prob > 0.9999) & (abs(df.log2FC) > 2)]

#selecting only the gene names and the GO associated

sub_df_degs= df_filtered[["transcript_id", "GO"]]

#removing the duplicated entries

sub_df_degs_uniq = sub_df_degs.drop_duplicates()
```

```
#selecting only the gene name, their GO and the orthologus gene
description in a new dataframe

sub_df_annot= annotation[["transcript_id", "GO",
"ortho_gene_description"]]

#removing the duplicated entries

sub_df_annot_uniq = sub_df_annot.drop_duplicates()


#creating a dictionary with the frequency of the OBSERVED GO among DEGs

GO_obs_dict = {}

for i in sub_df_degs["GO"].astype(str):

    for j in i.split("; "):

        if j in GO_obs_dict:

            GO_obs_dict[j] += 1

        else:

            GO_obs_dict[j] = 1

#converting the dictionary to a dataframe

GO_obs = pd.DataFrame.from_dict(GO_obs_dict, orient="index",
columns=["observed_counts"])


#creating a dictionary with the frequency of the TOTAL GO among annotated
genes

GO_total_dict = {}

for i in sub_df_annot_uniq["GO"].astype(str):

    for j in i.split("; "):

        if j in GO_total_dict:

            GO_total_dict[j] += 1

        else:

            GO_total_dict[j] = 1

#converting the dictionary to a dataframe

GO_tot = pd.DataFrame.from_dict(GO_total_dict, orient="index",
columns=["total_counts"])


#concatenating both dataframes previously produced
```

```
GO_test = GO_tot.merge(GO_obs, how = "outer", left_index=True,
right_index=True)

#discarding the NAs

GO_test["observed_counts"][GO_test.observed_counts.isna()] = 0



#creating two variables correspondent to the sum of the total counts of
the OBSERVED and TOTAL GOs

N = GO_test.total_counts.sum()

n = GO_test.observed_counts.sum()



#hypergeometric test between the OSERVED and the TOTAL amount of GOs

GO_test["pvalue"] = hypergeom.sf(GO_test.observed_counts, N,
GO_test.total_counts, n)

GO_test["obs-expect"] = GO_test["observed_counts"] - (n *
GO_test["total_counts"])/N



#filtering the test results in a list

a = list(GO_test[(GO_test.pvalue < 0.0005) & (GO_test["obs-expect"] >
3)].index)



#removing the NAs from the annotation dataframe (previously reduced to
few columns)

sub_df_annot_uniq = sub_df_annot_uniq.fillna("")



#creating a boxplot for each GO which passed the filtering parameters

for i in a:

    id_interessanti =
list(sub_df_annot_uniq.transcript_id[sub_df_annot_uniq["GO"].str.contains
(i)])

    tpm_filt= tpm.loc[id_interessanti]

    sums = pd.DataFrame(tpm_filt.sum(axis=0), columns=["total_tpm"])

    sums["cat"] = ['C', 'C', 'C', 'H','H','H']

    sums.boxplot(by="cat")

    plt.savefig('/home/CC/' + i + '.png')
```

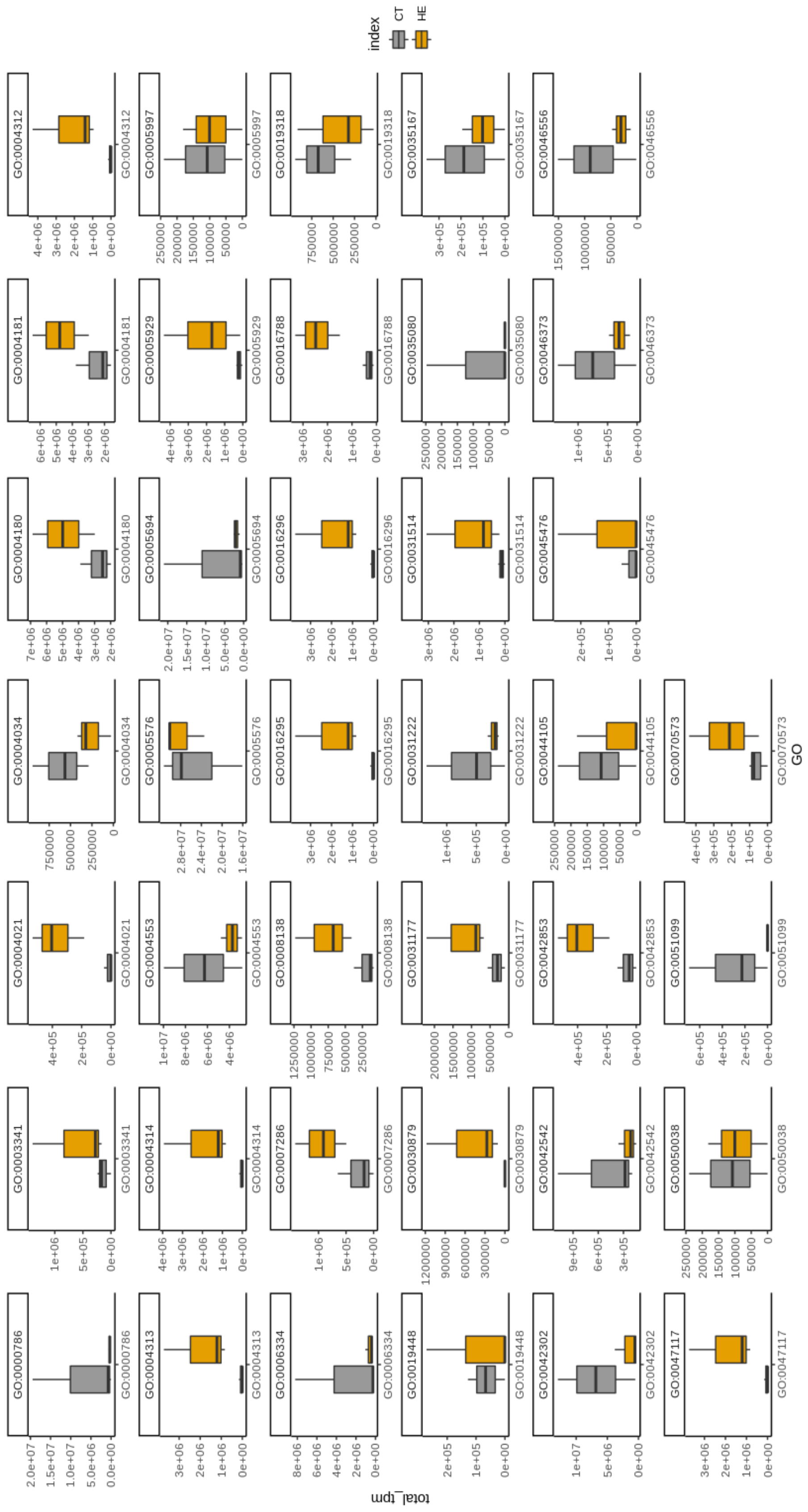
Supplemental Protocol S3. PFAM enrichment analysis performed with python3.7

```
import pandas as pd

import numpy as np

from scipy.stats import hypergeom

import matplotlib.pyplot as plt




#importing the transcript annotation table previously obtained with a custom python script

annotation = pd.read_csv("/home/CC/Cryptopygus_transcript_annotation.tsv", sep = "\t")

#importing the DEGs table previously obtained with the NOISeq package

degs = pd.read_csv("/home/CC/DEGs.csv", sep= ",")

degs.rename(columns={'Unnamed: 0':'transcript_id'}, inplace=True)

#importing the gene expression table previously obtained with SALMON

tpm = pd.read_csv("/home/CC/Cryptopygus_gene_expression.tsv", sep = "\t")

tpm.index = tpm.Name

tpm = tpm[["Ha", "Hb", "Hc", "SSC1a", "SSC1b", "SSC1c", ]]




#merging the annotation dataframe with the DEGs

df = degs.merge(annotation, on = "transcript_id", how = "left")

#filtering the resulting dataframe by a cutoff probability and their log2fold-change

df_filtered = df[(df.prob > 0.9999) & (abs(df.log2FC) > 2)]

#selecting only the gene names and the pfam associated

sub_df_degs= df_filtered[["transcript_id", "pfam"]]

#removing the duplicated entries

sub_df_degs_uniq = sub_df_degs.drop_duplicates()
```

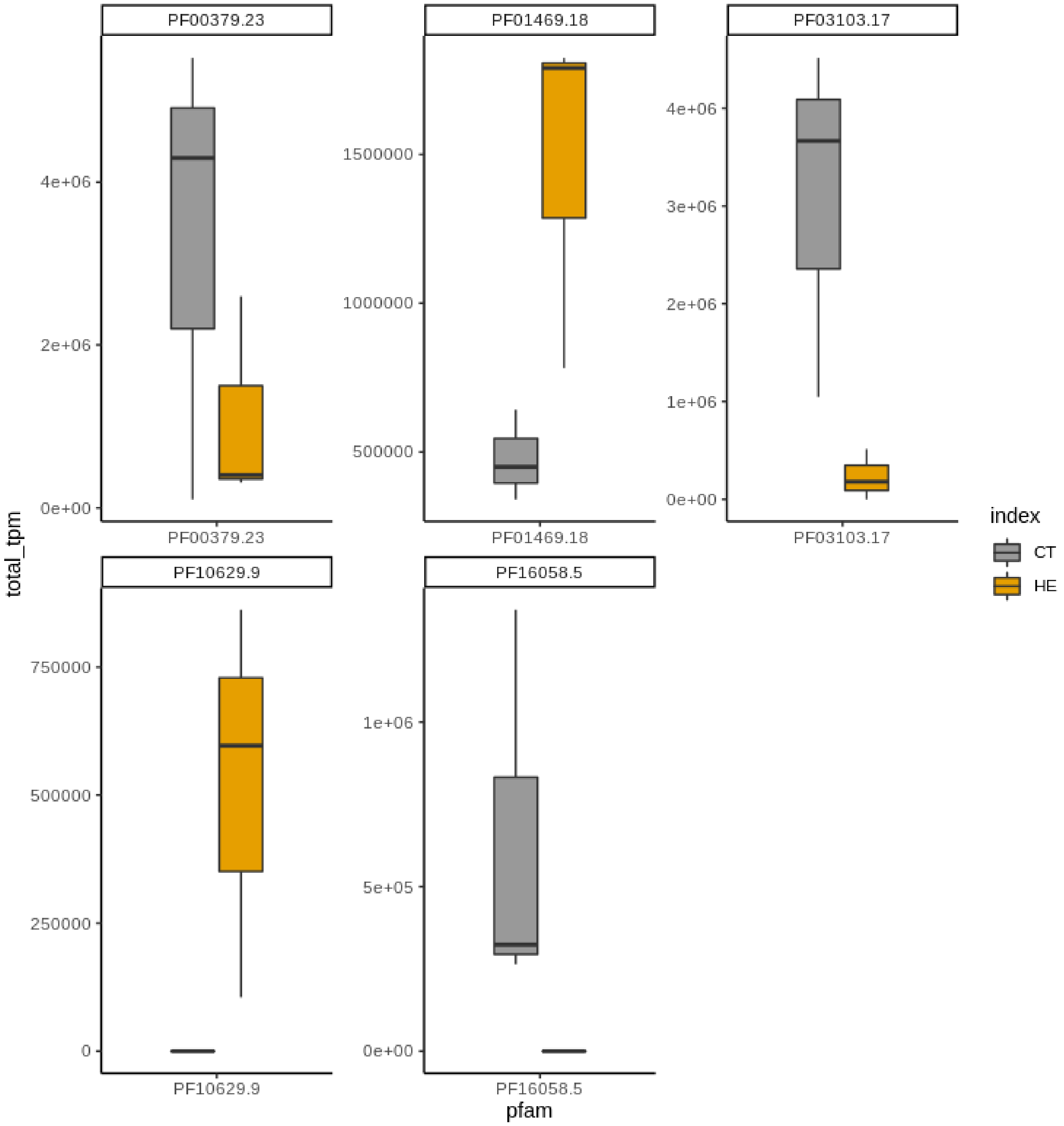
```python
#selecting only the gene name, their pfam and the orthologus gene description in a new dataframe
sub_df_annot= annotation[["transcript_id", "pfam", "ortho_gene_description"]]
#removing the duplicated entries
sub_df_annot_uniq = sub_df_annot.drop_duplicates()


#creating a dictionary with the frequency of the OBSERVED pfam among DEGs
pfam_obs_dict = {}
for i in sub_df_degs["pfam"].astype(str):
    for j in i.split("; "):
        if j in pfam_obs_dict:
            pfam_obs_dict[j] += 1
        else:
            pfam_obs_dict[j] = 1
#converting the dictionary to a dataframe
pfam_obs = pd.DataFrame.from_dict(pfam_obs_dict, orient="index", columns=["observed_counts"])


#creating a dictionary with the frequency of the TOTAL pfam among annotated genes
pfam_total_dict = {}
for i in sub_df_annot_uniq["pfam"].astype(str):
    for j in i.split("; "):
        if j in pfam_total_dict:
            pfam_total_dict[j] += 1
        else:
            pfam_total_dict[j] = 1
#converting the dictionary to a dataframe
pfam_tot = pd.DataFrame.from_dict(pfam_total_dict, orient="index", columns=["total_counts"])


#concatenating both dataframes previously produced
```

```
pfam_test = pfam_tot.merge(pfam_obs, how = "outer", left_index=True, right_index=True)

#discarding the NAs

pfam_test["observed_counts"][pfam_test.observed_counts.isna()] = 0


#creating two variables correspondent to the sum of the total counts of the OBSERVED and TOTAL pfams

N = pfam_test.total_counts.sum()

n = pfam_test.observed_counts.sum()


#hypergeometric test between the OSERVED and the TOTAL amount of pfams

pfam_test["pvalue"] = hypergeom.sf(pfam_test.observed_counts, N, pfam_test.total_counts, n)

pfam_test["obs-expect"] = pfam_test["observed_counts"] - (n * pfam_test["total_counts"])/N


#filtering the test results in a list

a = list(pfam_test[(pfam_test.pvalue < 0.0005) & (pfam_test["obs-expect"] > 3)].index)


#removing the NAs from the annotation dataframe (previously reduced to few columns)

sub_df_annot_uniq = sub_df_annot_uniq.fillna("")


#creating a boxplot for each pfam which passed the filtering parameters

for i in a:

    id_interessanti = list(sub_df_annot_uniq.transcript_id[sub_df_annot_uniq["pfam"].str.contains(i)])

    tpm_filt= tpm.loc[id_interessanti]

    sums = pd.DataFrame(tpm_filt.sum(axis=0), columns=["total_tpm"])

    sums["cat"] = ['C', 'C', 'C', 'H','H','H']

    sums.boxplot(by="cat")

    plt.savefig('/home/CC/' + i + '.png')
```

Supplemental Figure S1. GO enrichment analysis boxplots. The two springtail group conditions studied in this work (i.e. CT= control - grey; HE=heated - orange) are represented as boxplots for each of the Gene Ontology term enriched through the bioinformatic analysis performed with Protocol S2.



Supplemental Figure S2. PFAM enrichment analysis boxplots. The two springtail group conditions studied in this work (i.e. CT= control - grey; HE=heated - orange) are represented as boxplots for each of the PFAM term enriched through the bioinformatic analysis performed with Protocol S3.



Supplemental Table S1. Number of reads per sample and sample ID for control (CT) and heated (HE) conditions

Label	Sample	Number of raw reads
CT1	Control_1	41,4 million
CT2	Control_2	16,8 million
CT3	Control_3	17,4 million
HE1	Heated_1	18,8 million
HE2	Heated_2	15,8 million
HE3	Heated_3	14,7 million

Supplementary Table S2. Summary of the transcript numbers processed during the functional analysis of *C. terranovus* transcriptome. HE= heated samples.

	Number of transcripts
Total transcriptome	64,039
Annotated transcriptome	45,303
Differential expressed genes	7,637
HE up-regulated genes	3,719
HE down-regulated genes	3,918
Annotated differentially expresse	5,425

Supplemental Table S3. Differential Expressed Genes in the HE (yellow highlighted) and CT conditions (blue highlighted) grouped by their Gene Ontology classification following the Uniprot annotation Columns: Class indicates the general GO family name; GO indicates the GO examined and found over-represented in one condition or the other; Unipro_ID stands for the unique AN related to the Uni prot database; Uniprot_name stands for the model organisms' protein name, where OS:

Transcripts enhanced in the Heated (HE) group

Class	GO	uniprot_id	UniProt_name (OS=Organism species; GN=Gene Name)
Protein degradation	GO:0070573	Q6GQ29	Carboxypeptidase Q OS=Xenopus laevis OX=8355 GN=cpq PE=2 SV=1
		Q3SZM7	Dipeptidase 1 OS=Bos taurus OX=9913 GN=DPEP1 PE=2 SV=1
		P12821	Angiotensin-converting enzyme OS=Homo sapiens OX=9606 GN=ACE PE=1 SV=1
	GO:0004021	Q8BGT5	Alanine aminotransferase 2 OS=Mus musculus OX=10090 GN=Gpt2 PE=1 SV=1
		A4IFH5	Alanine aminotransferase 1 OS=Bos taurus OX=9913 GN=GPT PE=2 SV=1
		Q6NYL5	Alanine aminotransferase 2-like OS=Danio rerio OX=7955 GN=gpt2l PE=2 SV=2
		P24298	Alanine aminotransferase 1 OS=Homo sapiens OX=9606 GN=GPT PE=1 SV=3
	GO:0004180	P15169	Carboxypeptidase N catalytic chain OS=Homo sapiens OX=9606 GN=CPN1 PE=1 SV=1
		Q6GQ29	Carboxypeptidase Q OS=Xenopus laevis OX=8355 GN=cpq PE=2 SV=1
		P12821	Angiotensin-converting enzyme OS=Homo sapiens OX=9606 GN=ACE PE=1 SV=1
		Q8C1W1	Tubulinyl-Tyr carboxypeptidase 1 OS=Mus musculus OX=10090 GN=Vash1 PE=1 SV=4
	GO:0004181	P15169	Carboxypeptidase N catalytic chain OS=Homo sapiens OX=9606 GN=CPN1 PE=1 SV=1
		Q8C1W1	Tubulinyl-Tyr carboxypeptidase 1 OS=Mus musculus OX=10090 GN=Vash1 PE=1 SV=4
	GO:0042853	A4IFH5	Alanine aminotransferase 1 OS=Bos taurus OX=9913 GN=GPT PE=2 SV=1
		P24298	Alanine aminotransferase 1 OS=Homo sapiens OX=9606 GN=GPT PE=1 SV=3
		Q6NYL5	Alanine aminotransferase 2-like OS=Danio rerio OX=7955 GN=gpt2l PE=2 SV=2
		Q8BGT5	Alanine aminotransferase 2 OS=Mus musculus OX=10090 GN=Gpt2 PE=1 SV=1
Spermatid development	GO:007286	Q09363	Galactosylgalactosylxylosylprotein 3-beta-glucuronosyltransferase sqv-8 OS=Caenorhabditis elegans OX=6239 GN=sqv-8 PE=1 SV=1
		Q96MC6	Hippocampus abundant transcript 1 protein OS=Homo sapiens OX=9606 GN=MFSD14A PE=1 SV=2
		O75602	Sperm-associated antigen 6 OS=Homo sapiens OX=9606 GN=SPAG6 PE=2 SV=1
		O75602	Sperm-associated antigen 6 OS=Homo sapiens OX=9606 GN=SPAG6 PE=2 SV=1
		Q29L39	Protein purity of essence OS=Drosophila pseudoobscura pseudoobscura OX=46245 GN=poe PE=3 SV=1
		Q8CDV6	Coiled-coil domain-containing protein 63 OS=Mus musculus OX=10090 GN=Ccdc63 PE=2 SV=2
Carbohydrate metabolim	GO:0004034	P40801	Bifunctional protein GAL10 OS=Pachysolen tannophilus OX=4918 GN=GAL10 PE=2 SV=1
		Q9GKX6	Aldose 1-epimerase OS=Sus scrofa OX=9823 GN=GALM PE=2 SV=1
	GO:0004553	Q08638	Beta-glucosidase A OS=Thermotoga maritima (strain ATCC 43589 / MSB8 / DSM 3109 / JCM 10099) OX=243274 GN=bglA PE=1 SV=1
		Q8S8Q4	Endoglucanase 14 OS=Arabidopsis thaliana OX=3702 GN=At2g44560 PE=2 SV=2
		P49010	Chitooligosaccharidolytic beta-N-acetylglucosaminidase OS=Bombyx mori OX=7091 PE=1 SV=1
		O97524	Beta-glucuronidase OS=Felis catus OX=9685 GN=GUSB PE=1 SV=1
		Q9NHB0	Gram-negative bacteria-binding protein 1 OS=Drosophila melanogaster OX=7227 GN=GNBP1 PE=1 SV=2
		Q8MU95	Beta-1,3-glucan-binding protein OS=Plodia interpunctella OX=58824 PE=1 SV=1
		Q90744	Alpha-N-acetylgalactosaminidase OS=Gallus gallus OX=9031 GN=NAGA PE=1 SV=1
		P26221	Endoglucanase E-4 OS=Thermobifida fusca OX=2021 GN=celD PE=1 SV=2
		P09849	Lactase-phlorizin hydrolase OS=Oryctolagus cuniculus OX=9986 GN=LCT PE=1 SV=1
		Q76FP5	Endo-beta-1,6-galactanase OS=Hypocrea rufa OX=5547 GN=6GAL PE=1 SV=1
Lipid metabolism	GO:0004312	P36189	Fatty acid synthase (Fragment) OS=Anser anser anser OX=8844 GN=FASN PE=1 SV=1
		P19096	Fatty acid synthase OS=Mus musculus OX=10090 GN=Fasn PE=1 SV=2
	GO:0004313	P19096	Fatty acid synthase OS=Mus musculus OX=10090 GN=Fasn PE=1 SV=2
	GO:0004314	P19096	Fatty acid synthase OS=Mus musculus OX=10090 GN=Fasn PE=1 SV=2
	GO:0016295	P19096	Fatty acid synthase OS=Mus musculus OX=10090 GN=Fasn PE=1 SV=2
	GO:0016296	P19096	Fatty acid synthase OS=Mus musculus OX=10090 GN=Fasn PE=1 SV=2
	GO:0016788	O46107	Lipase 1 OS=Drosophila melanogaster OX=7227 GN=Lip1 PE=2 SV=2
		P19096	Fatty acid synthase OS=Mus musculus OX=10090 GN=Fasn PE=1 SV=2
		Q5VXJ0	Lipase member K OS=Homo sapiens OX=9606 GN=LIPK PE=2 SV=2
		Q91V76	Ester hydrolase C11orf54 homolog OS=Mus musculus OX=10090 PE=1 SV=1
	GO:0031177	Q4WLD4	Non-reducing polyketide synthase pyr2 OS=Neosartorya fumigata (strain ATCC MYA-4609 / Af293 / CBS 101355 / FGSC A1100) OX=330879 GN=pyr2 PE=1 SV=2
		Q03131	6-deoxyerythronolide-B synthase EryA1, modules 1 and 2 OS=Saccharopolyspora erythraea OX=1836 GN=eryA PE=1 SV=1
		P19096	Fatty acid synthase OS=Mus musculus OX=10090 GN=Fasn PE=1 SV=2
	GO:00471177	P19096	Fatty acid synthase OS=Mus musculus OX=10090 GN=Fasn PE=1 SV=2

Transcripts enhanced in the Control (CT) group

Class	GO	uniprot_id	UniProt_name (OS=Organism species; GN=Gene Name)
Protein degradation	GO:0004021	Q8TD30	Alanine aminotransferase 2 OS=Homo sapiens OX=9606 GN=GPT2 PE=1 SV=1
	GO:0004180	P04069	Carboxypeptidase B OS=Astacus astacus OX=6715 PE=1 SV=1
		Q80V42	Carboxypeptidase M OS=Mus musculus OX=10090 GN=Cpm PE=1 SV=2
		Q3ZC84	Cytosolic non-specific dipeptidase OS=Bos taurus OX=9913 GN=CNDP2 PE=2 SV=1
		Q5RFE4	Probable serine carboxypeptidase CPVL OS=Pongo abelii OX=9601 GN=CPVL PE=2 SV=1
		Q2KIG3	Carboxypeptidase B2 OS=Bos taurus OX=9913 GN=CPB2 PE=1 SV=1
		Q92I71	Putative carboxypeptidase RC0549 OS=Rickettsia conorii (strain ATCC VR-613 / Malish 7) OX=272944 GN=RC0549 PE=3 SV=2
	GO:0004181	Q2KIG3	Carboxypeptidase B2 OS=Bos taurus OX=9913 GN=CPB2 PE=1 SV=1
		Q80V42	Carboxypeptidase M OS=Mus musculus OX=10090 GN=Cpm PE=1 SV=2
		P04069	Carboxypeptidase B OS=Astacus astacus OX=6715 PE=1 SV=1
	GO:0042853	P41689	Serine--pyruvate aminotransferase, mitochondrial OS=Felis catus OX=9685 GN=AGXT PE=2 SV=1
		P41689	Serine--pyruvate aminotransferase, mitochondrial OS=Felis catus OX=9685 GN=AGXT PE=2 SV=1
		Q8TD30	Alanine aminotransferase 2 OS=Homo sapiens OX=9606 GN=GPT2 PE=1 SV=1
Spermatid development	GO:007286	Q9Y385	Ubiquitin-conjugating enzyme E2 J1 OS=Homo sapiens OX=9606 GN=UBE2J1 PE=1 SV=2
		Q8UVX0	Piwi-like protein 1 OS=Danio rerio OX=7955 GN=piwil1 PE=1 SV=1
		Q9Y232	Chromodomain Y-like protein OS=Homo sapiens OX=9606 GN=CDYL PE=1 SV=2
		Q6JEL2	Kelch-like protein 10 OS=Homo sapiens OX=9606 GN=KLHL10 PE=1 SV=1
Carbohydrate metabolim	GO:0004034	Q96C23	Aldose 1-epimerase OS=Homo sapiens OX=9606 GN=GALM PE=1 SV=1
	GO:0004553	P82594	Extracellular exo-alpha-(1->5)-L-arabinofuranosidase OS=Streptomyces chartreusis OX=1969 PE=1 SV=1
		O43451	Maltase-glucoamylase, intestinal OS=Homo sapiens OX=9606 GN=MGAM PE=1 SV=5
		P36362	Endochitinase OS=Manduca sexta OX=7130 PE=2 SV=1
		P04067	Endo-beta-N-acetylglucosaminidase H OS=Streptomyces plicatus OX=1922 PE=1 SV=1
		Q8IW92	Beta-galactosidase-1-like protein 2 OS=Homo sapiens OX=9606 GN=GLB1L2 PE=2 SV=1
		O62653	Sucrase-isomaltase, intestinal OS=Suncus murinus OX=9378 GN=SI PE=2 SV=3
		Q95X01	Myrosinase 1 OS=Brevicoryne brassicae OX=69196 PE=1 SV=1
		P08236	Beta-glucuronidase OS=Homo sapiens OX=9606 GN=GUSB PE=1 SV=2
		Q8NCI6	Beta-galactosidase-1-like protein 3 OS=Homo sapiens OX=9606 GN=GLB1L3 PE=2 SV=3
		Q0DA21	Beta-glucosidase 25 OS=Oryza sativa subsp. japonica OX=39947 GN=BGLU25 PE=2 SV=2
		Q69ZQ1	Myogenesis-regulating glycosidase OS=Mus musculus OX=10090 GN=Myorg PE=1 SV=2
		Q25BW5	Beta-glucosidase 1A OS=Phanerochaete chrysosporium OX=5306 GN=BGL1A PE=1 SV=1
		Q0CMA7	Probable alpha/beta-glucosidase agdC OS=Aspergillus terreus (strain NIH 2624 / FGSC A1156) OX=341663 GN=agdC PE=3 SV=1
		Q54KX5	Protein-glucosylgalactosylhydroxylysine glucosidase OS=Dictyostelium discoideum OX=44689 GN=pgghg PE=3 SV=2
		Q0V8R6	Beta-hexosaminidase subunit alpha OS=Bos taurus OX=9913 GN=HEXA PE=2 SV=1
		G4MMH2	Alpha-L-arabinofuranosidase B OS=Magnaporthe oryzae (strain 70-15 / ATCC MYA-4617 / FGSC 8958) OX=242507 GN=abfB PE=1 SV=1
		Q8BHN3	Neutral alpha-glucosidase AB OS=Mus musculus OX=10090 GN=Ganab PE=1 SV=1
	GO:0046373	Q0CTV2	Probable alpha-L-arabinofuranosidase A OS=Aspergillus terreus (strain NIH 2624 / FGSC A1156) OX=341663 GN=abfA PE=3 SV=1
		G4MMH2	Alpha-L-arabinofuranosidase B OS=Magnaporthe oryzae (strain 70-15 / ATCC MYA-4617 / FGSC 8958) OX=242507 GN=abfB PE=1 SV=1
		Q96X54	Probable alpha-L-arabinofuranosidase A OS=Aspergillus awamori OX=105351 GN=abfA PE=1 SV=1
	GO:0046556	Q0CTV2	Probable alpha-L-arabinofuranosidase A OS=Aspergillus terreus (strain NIH 2624 / FGSC A1156) OX=341663 GN=abfA PE=3 SV=1
		G4MMH2	Alpha-L-arabinofuranosidase B OS=Magnaporthe oryzae (strain 70-15 / ATCC MYA-4617 / FGSC 8958) OX=242507 GN=abfB PE=1 SV=1
		Q96X54	Probable alpha-L-arabinofuranosidase A OS=Aspergillus awamori OX=105351 GN=abfA PE=1 SV=1
		P82594	Extracellular exo-alpha-(1->5)-L-arabinofuranosidase OS=Streptomyces chartreusis OX=1969 PE=1 SV=1
Lipid metabolism	GO:0016788	P30122	Bile salt-activated lipase (Fragment) OS=Bos taurus OX=9913 GN=CEL PE=1 SV=2
	GO:0031177	Q7TXL8	Phthiocerol/phenolphthiocerol synthesis polyketide synthase type I PpsC OS=Mycobacterium bovis (strain ATCC BAA-935 / AF2122/97) OX=233413 GN=ppsC PE=1 SV=1
		A0R1E8	Mycocerosic acid synthase-like polyketide synthase OS=Mycobacterium smegmatis (strain ATCC 700084 / mc(2)155) OX=246196 GN=pks5 PE=1 SV=1
		Q50LG3	Highly reducing polyketide synthase AFT9-1 OS=Alternaria alternata OX=5599 GN=AFT9-1 PE=3 SV=1