

title: “Sunflower Rhythms 2020 Post-COSOPT Analysis” output: pdf_document: default html_notebook: default html_document: df_print: paged —

Setup the R environment

```
library(circular)

##
## Attaching package: 'circular'
## The following objects are masked from 'package:stats':
##
##      sd, var
library(clockplot)
library(ggplot2)
library(reshape2)
library(plyr)
library(stringr)
library(tools)
library(VennDiagram)

## Loading required package: grid
## Loading required package: futile.logger
knitr::opts_knit$set(root.dir='.')
```

Set thresholds and colors

```
min.p.mmc.beta <- 0.05
min.meanexplev <- 0.2
per.buffer <- 2
exp.min <- 10
amp.min <- 0.2
east.color <- 'orange'
west.color <- 'forestgreen'
```

Import and pre-process time course data

```
if (!file.exists('counts/east-counts.tsv')
    | !file.exists('counts/west-counts.tsv')
    | !file.exists('r-data/timecourse.rds')) {

  if (!dir.exists('r-data')) dir.create('r-data')

  counts <- read.table('counts/reanalysis_HA2015_HanXRQr1.0_mRNA_normalized_arranged.csv', sep=',', row
    # Remove bad replicates
    counts <- counts[, ! colnames(counts) %in% c('X4ea2', 'X10ea3', 'X16ea3', 'X10w3', 'X15w2')]

  # Extract sample side from column names
  west.samples <- grepl('w', colnames(counts))
  east.samples <- grepl('e', colnames(counts))
```

```

side <- rep('', length(colnames(counts)))
side[west.samples] <- 'West'
side[east.samples] <- 'East'

saveRDS(side, 'r-data/side.rds')

# Extract Zeitgeber Time from column names
time.idx <- as.integer(sub("X([0-9]+)[ew][ae]?[1-3]{1}", "\\1", colnames(counts)))
times <- seq(0, 46, 2)
hour <- times[time.idx]
saveRDS(hour, 'r-data/hour.rds')

# Prepare timecourse for plotting
timecourse <- data.frame(hour, side, t(counts))
timecourse <- melt(timecourse, id.vars=c('hour', 'side'), variable.name='gene', value.name='counts', na.rm=TRUE)
timecourse <- ddpby(timecourse, .(hour, side, gene), summarize, mean=mean(counts), stderr=sqrt(var(counts)))
saveRDS(timecourse, 'r-data/timecourse.rds')

# Output East and West counts files
saveRDS(counts, 'r-data/counts.rds')

counts[] <- lapply(counts, as.character)
counts <- rbind(hour, counts)
rownames(counts)[1] <- 'Gene'

west.counts <- counts[, west.samples]
east.counts <- counts[, east.samples]

write.table(east.counts, 'counts/east-counts.tsv', sep='\t', quote=F, col.names=F)
write.table(west.counts, 'counts/west-counts.tsv', sep='\t', quote=F, col.names=F)

saveRDS(east.counts, 'r-data/east.counts.rds')
saveRDS(west.counts, 'r-data/west.counts.rds')
}

if(!exists("timecourse")) timecourse <- readRDS('r-data/timecourse.rds')

```

Function to plot timecourse data and demo

```

if (!dir.exists('plots')) dir.create('plots')

plot.timecourse <- function(gene.list, east.color='orange', west.color='forestgreen',
                             double.plot=FALSE, side.by.side=FALSE, backlit=TRUE, theme.bw=TRUE,
                             lights.off=NULL, custom.daynight=NULL, night.alpha=0.7,
                             print.plot=TRUE, return.plot=FALSE) {
  library(ggplot2)
  timecourse.subset <- timecourse[timecourse$gene %in% gene.list, ]
  timecourse.subset$gene <- as.character(timecourse.subset$gene)

  if (double.plot) {
    timecourse.subset.copy <- timecourse.subset
    timecourse.subset.copy$hour <- timecourse.subset.copy$hour + 48
  }

```

```

timecourse.subset <- rbind(timecourse.subset, timecourse.subset.copy)
x.breaks <- seq(0, 96, 12)
} else {
  x.breaks <- seq(0, 48, 12)
}

p <- ggplot()

daynight <- NULL
if(!is.null(custom.daynight)) {
  # Example of custom.daynight:
  # data.frame(dawn=c(0, 24, 48, 72, 96), dusk=c(13.25 - 24, 13.25, 13.25 + 24, 13.25 + 48, 13.25 + 72))
  daynight <- custom.daynight
} else if (!is.null(lights.off)) {
  lights.on <- seq(floor(min(timecourse.subset$hour) / 24), 24 * ceiling(max(timecourse.subset$hour) / 24))
  daynight <- data.frame(dawn=lights.on, dusk=lights.on + lights.off %% 24 - 24)
}

if (!is.null(daynight)) {
  p <- p + geom_rect(data=daynight, aes(xmin=dawn, xmax=dusk), fill="black", ymin=-10000, ymax=10000,
}

if (backlit) {
  p <- p +
    geom_line(data=subset(timecourse.subset, side=='West'), aes(x=hour, y=mean), color='white', size=1) +
    geom_line(data=subset(timecourse.subset, side=='East'), aes(x=hour, y=mean), color='white', size=1) +
    geom_errorbar(data=subset(timecourse.subset, side=='West'), aes(x=hour, ymin=mean-stderr, ymax=mean+stderr), color='white', size=1) +
    geom_errorbar(data=subset(timecourse.subset, side=='East'), aes(x=hour, ymin=mean-stderr, ymax=mean+stderr), color='white', size=1)
}

p <- p +
  geom_line(data=timecourse.subset, aes(x=hour, y=mean, color=side), size=1) +
  geom_line(data=timecourse.subset, aes(x=hour, y=mean, color=side), size=1) +
  geom_errorbar(data=timecourse.subset, aes(x=hour, color=side, ymin=mean-stderr, ymax=mean+stderr), size=1) +
  labs(x = 'Time (hours)', y = 'Mean Normalized Counts') +
  scale_x_continuous(breaks=x.breaks) +
  scale_color_manual(name='Side', values=c(east.color, west.color))

if (double.plot) {
  p <- p + coord_cartesian(xlim=c(0, 96), expand=T)
} else {
  p <- p + coord_cartesian(xlim=c(0, 48), expand=T)
}

if (side.by.side) {
  p <- p + facet_grid(gene ~ side, scales='free_y')
} else {
  p <- p + facet_wrap(~ gene, ncol=1, scales='free_y')
}

if (theme.bw) {
  p <- p + theme_bw() + theme(strip.background = element_rect(fill='white'))
}

```

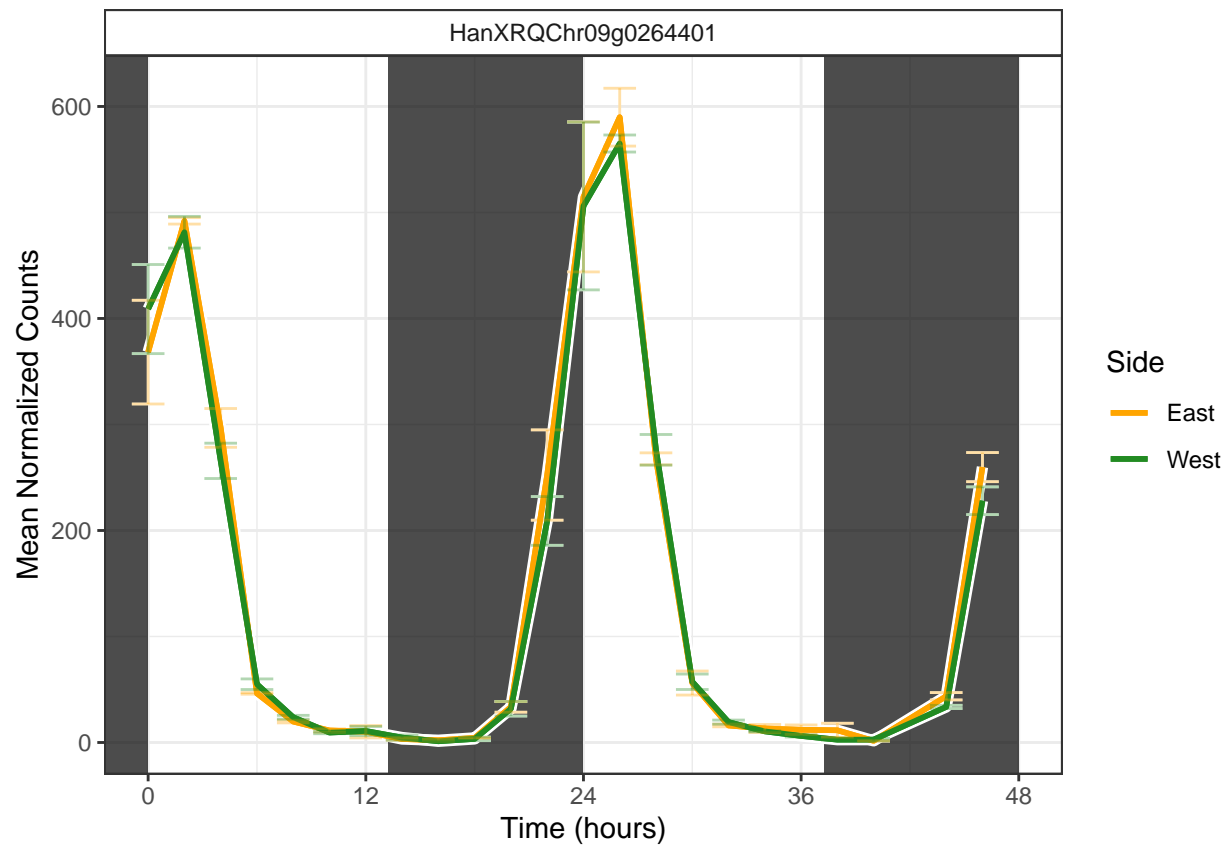
```

    if (print.plot) print(p)
    if (return.plot) p
  }

demo.gene.list <- c('HanXRQChr09g0264401', 'HanXRQChr15g0489581', 'HanXRQChr04g0118841', 'HanXRQChr01g0118841')

# Plot single gene
plot.timecourse(demo.gene.list[1], lights.off=13.25)

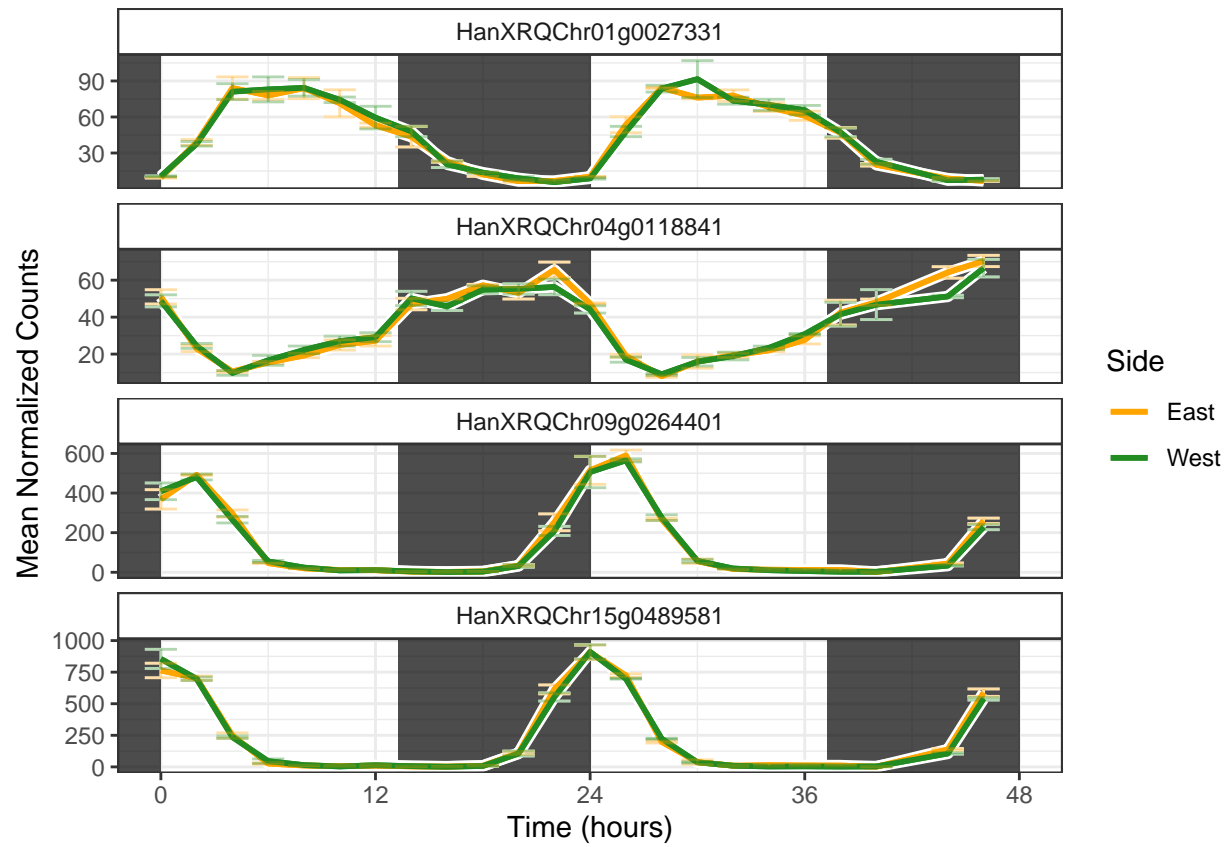
```



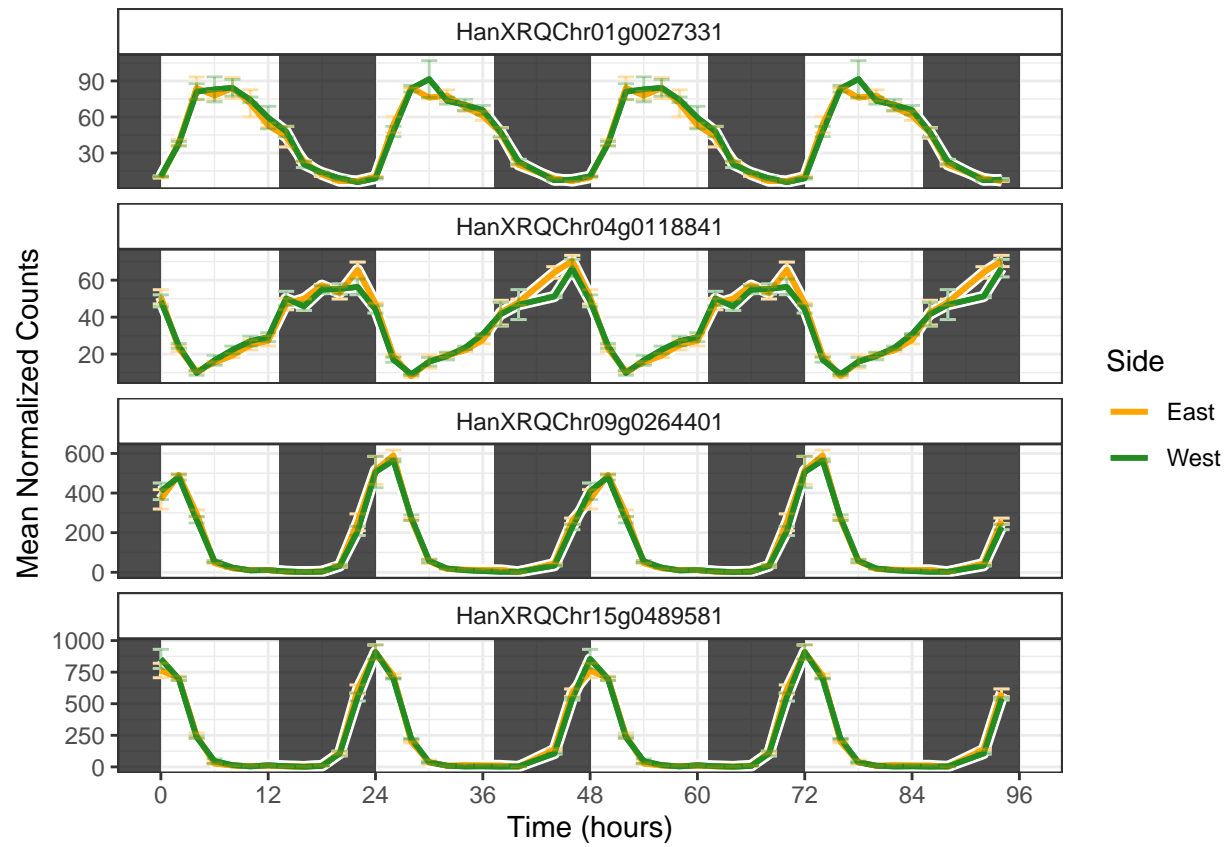
```

# Plot gene list
plot.timecourse(demo.gene.list, lights.off=13.25)

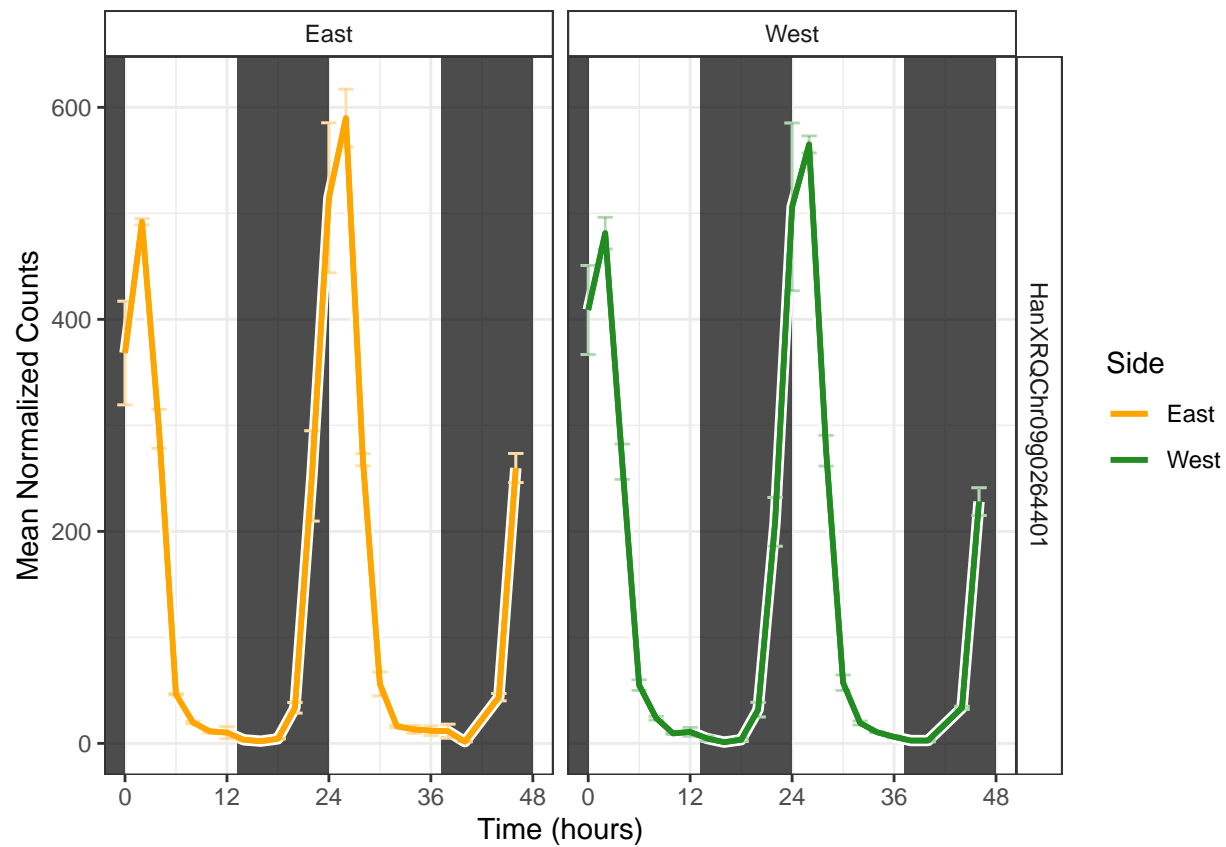
```



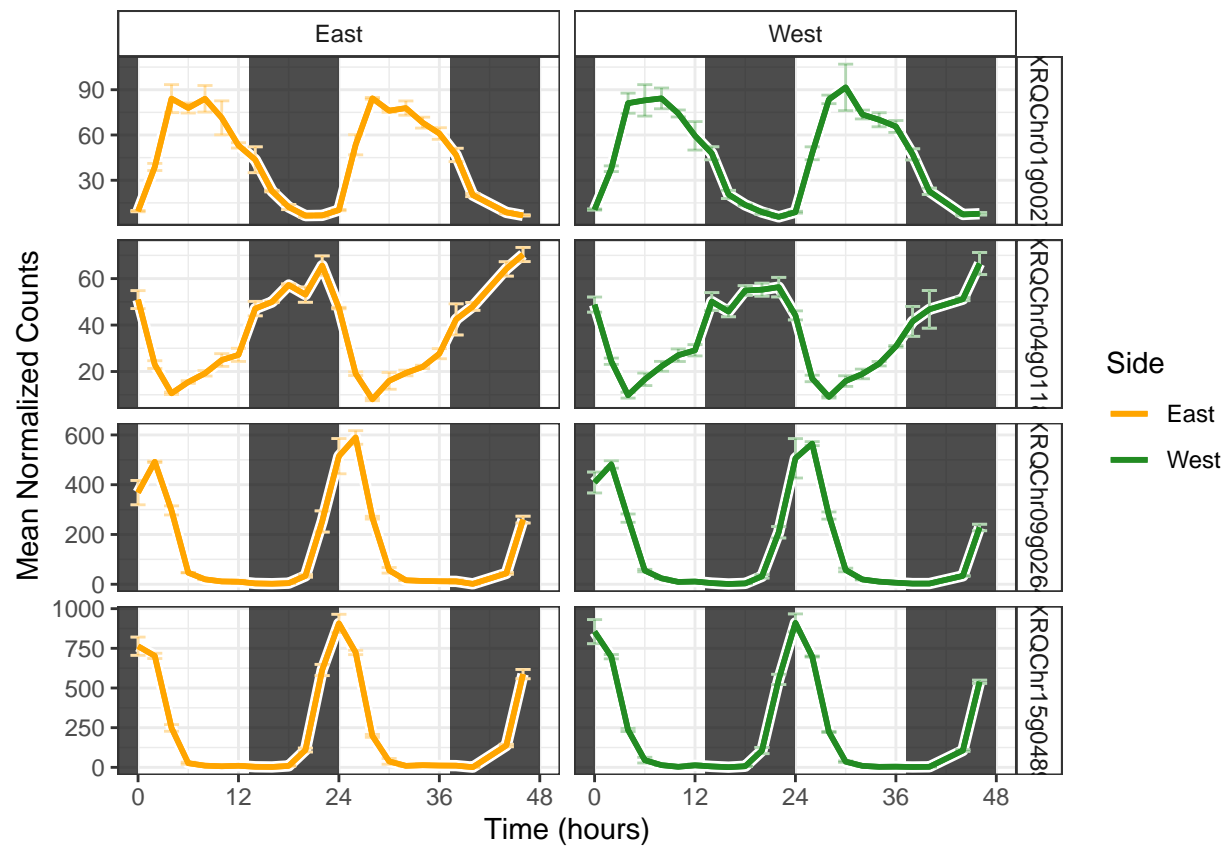
```
plot.timecourse(demo.gene.list, double.plot=TRUE, lights.off=13.25)
```



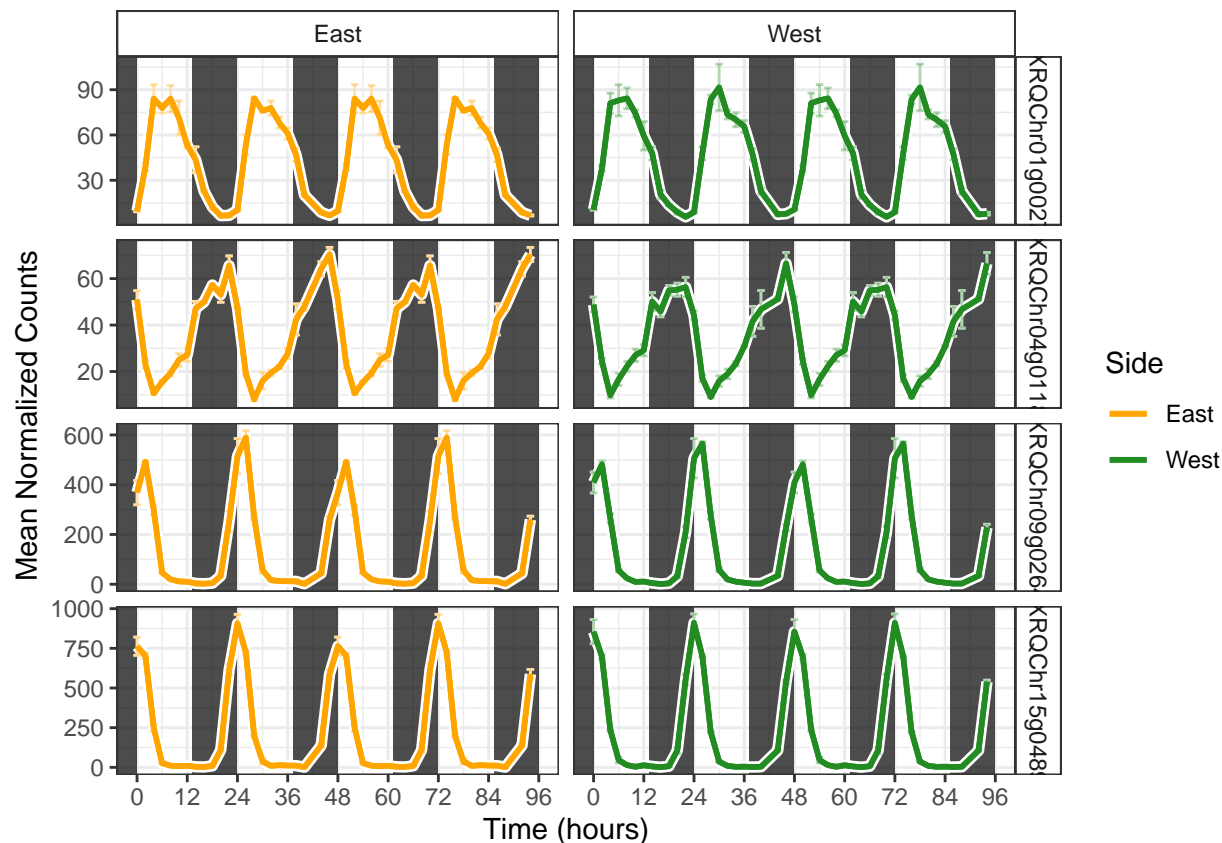
```
# Plot side-by-side
plot.timecourse(demo.gene.list[1], lights.off=13.25, side.by.side=TRUE)
```



```
plot.timecourse(demo.gene.list, lights.off=13.25, side.by.side=TRUE)
```



```
plot.timecourse(demo.gene.list, double.plot=TRUE, lights.off=13.25, side.by.side=TRUE)
```

Import COSOPT results and calculate additional metrics

We start with the COSOPT results files. They should have the following MD5 checksums:

```
4529c38ab3f52eb790416515f92774c3 cosopt/output-files/HA2015_HanXRQr1.0-East.cosopt-results.tsv
756c59834b09b678d05d4758bc995673 cosopt/output-files/HA2015_HanXRQr1.0-Merged.cosopt-results.tsv
f39d7991e9e917238172fd96d99bc38a cosopt/output-files/HA2015_HanXRQr1.0-West.cosopt-results.tsv
```

```
md5sum(list.files('cosopt/output-files', pattern='.tsv', full.names=TRUE))
```

```
## cosopt/output-files/HA2015_HanXRQr1.0-East.cosopt-results.tsv
## "4529c38ab3f52eb790416515f92774c3"
## cosopt/output-files/HA2015_HanXRQr1.0-Merged.cosopt-results.tsv
## "756c59834b09b678d05d4758bc995673"
## cosopt/output-files/HA2015_HanXRQr1.0-West.cosopt-results.tsv
## "f39d7991e9e917238172fd96d99bc38a"
```

```
if (!dir.exists('cosopt-processed')) dir.create('cosopt-processed')
```

```
cosopt.east <- read.table('cosopt/output-files/HA2015_HanXRQr1.0-East.cosopt-results.tsv', h=T)
cosopt.merged <- read.table('cosopt/output-files/HA2015_HanXRQr1.0-Merged.cosopt-results.tsv', h=T)
cosopt.west <- read.table('cosopt/output-files/HA2015_HanXRQr1.0-West.cosopt-results.tsv', h=T)
```

```
cosopt.east$RelAmp <- cosopt.east$Beta / cosopt.east$MeanExpLev
cosopt.west$RelAmp <- cosopt.west$Beta / cosopt.west$MeanExpLev
cosopt.merged$RelAmp <- cosopt.merged$Beta / cosopt.merged$MeanExpLev
```

```

cosopt.east$PeakPhase <- ifelse(cosopt.east$Phase <= 0, -cosopt.east$Phase, cosopt.east$Period - cosopt.east$Phase)
cosopt.west$PeakPhase <- ifelse(cosopt.west$Phase <= 0, -cosopt.west$Phase, cosopt.west$Period - cosopt.west$Phase)
cosopt.merged$PeakPhase <- ifelse(cosopt.merged$Phase <= 0, -cosopt.merged$Phase, cosopt.merged$Period - cosopt.merged$Phase)

cosopt.east$PeakPhase[cosopt.east$PeakPhase >= 24] <- cosopt.east$PeakPhase[cosopt.east$PeakPhase >= 24]
cosopt.west$PeakPhase[cosopt.west$PeakPhase >= 24] <- cosopt.west$PeakPhase[cosopt.west$PeakPhase >= 24]
cosopt.merged$PeakPhase[cosopt.merged$PeakPhase >= 24] <- cosopt.merged$PeakPhase[cosopt.merged$PeakPhase >= 24]

cosopt <- merge(cosopt.west, cosopt.east, by = 'GeneID', all = TRUE, suffixes = c('.W', '.E'))
cosopt <- merge(cosopt, cosopt.merged, by = 'GeneID', all = TRUE)

cosopt <- cosopt[, order(names(cosopt))]
rownames(cosopt) <- cosopt$GeneID

cosopt$phase.diff <- ifelse(
  abs(cosopt$PeakPhase.W - cosopt$PeakPhase.E) <= 12,
  cosopt$PeakPhase.W - cosopt$PeakPhase.E,
  ifelse(
    cosopt$PeakPhase.W - cosopt$PeakPhase.E < 0,
    cosopt$PeakPhase.W - cosopt$PeakPhase.E + 24,
    cosopt$PeakPhase.W - cosopt$PeakPhase.E - 24))

cosopt$amp.diff <- cosopt$RelAmp.W - cosopt$RelAmp.E

cosopt$exp.diff.log2 <- log(cosopt$MeanExpLev.W / cosopt$MeanExpLev.E, 2)

cosopt.processed.file <- 'cosopt-processed/cosopt-processed.txt'
write.table(cosopt, cosopt.processed.file, sep = "\t", quote = FALSE, col.names=NA)

# Expressed Genes
#Expressed in East or West: 33,188
nrow(subset(cosopt, MeanExpLev.E >= min.meanexplev | MeanExpLev.W >= min.meanexplev))

## [1] 33188

#Expressed in East and West: 26,928
nrow(subset(cosopt, MeanExpLev.E >= min.meanexplev & MeanExpLev.W >= min.meanexplev))

## [1] 26928

#Expressed in East: 30,166
nrow(subset(cosopt, MeanExpLev.E >= min.meanexplev))

## [1] 30166

#Expressed in West: 29,950
nrow(subset(cosopt, MeanExpLev.W >= min.meanexplev))

## [1] 29950

#Expressed in Merged: 30,844
nrow(subset(cosopt, MeanExpLev >= min.meanexplev))

## [1] 30844

```

```

# Get rhythmic genes
rhythmic.east <- as.character(cosopt.east$GeneID[cosopt.east$pMMC.Beta < min.p.mmc.beta & cosopt.east$M
rhythmic.west <- as.character(cosopt.west$GeneID[cosopt.west$pMMC.Beta < min.p.mmc.beta & cosopt.west$M
rhythmic.both <- intersect(rhythmic.east, rhythmic.west)
rhythmic.merged <- as.character(cosopt.merged$GeneID[cosopt.merged$pMMC.Beta < min.p.mmc.beta & cosopt.m
rhythmic.all <- intersect(rhythmic.both, rhythmic.merged)

length(intersect(rhythmic.merged, rhythmic.east))

## [1] 21391

# [1] 21605
length(intersect(rhythmic.merged, rhythmic.west))

## [1] 21366

# [1] 21585

rhythmic.east.only <- setdiff(rhythmic.east, rhythmic.both)
rhythmic.west.only <- setdiff(rhythmic.west, rhythmic.both)

length(rhythmic.east)

## [1] 22328

# [1] 22559
length(rhythmic.west)

## [1] 22374

# [1] 22623
length(rhythmic.merged)

## [1] 24574

# [1] 24914

length(rhythmic.both)

## [1] 19095

# [1] 19235
length(rhythmic.all)

## [1] 19062

# [1] 19201

length(rhythmic.east.only)

## [1] 3233

# [1] 3324
length(rhythmic.west.only)

## [1] 3279

# [1] 3388

```

```

if (!dir.exists('rhythmic-genes')) dir.create('rhythmic-genes')
write.table(sort(rhythmic.east), "rhythmic-genes/rhythmic-east.txt", sep = "\t", quote = FALSE, col.names = TRUE)
write.table(sort(rhythmic.west), "rhythmic-genes/rhythmic-west.txt", sep = "\t", quote = FALSE, col.names = TRUE)
write.table(sort(rhythmic.merged), "rhythmic-genes/rhythmic-merged.txt", sep = "\t", quote = FALSE, col.names = TRUE)

```

Rhythmic Counts Summary:

Total # of Genes: 49,262

Total # of Genes with at least one set of COSOPT results: 44,477

Total # of Expressed Genes:

East: 30,166

West: 29,950

East or West: 33,188

East and West: 26,928

Merged: 30,844

Rhythmic Genes in East and West time courses: 25,607

East only: 3,233 (12.6%)

West only: 3,279 (12.8%)

Both East and West: 19,095 (74.6%)

Rhythmic Genes in Merged time course: 24,574

Rhythmic Genes in all three time courses (East, West, and Merged): 19,062

Venn Diagram of Rhythmic Genes

```

threeway.Venn <- function(A, B, C, cat.names = c("A", "B", "C")){
  area1 <- length(A)
  area2 <- length(B)
  area3 <- length(C)
  n12 <- length(intersect(A,B))
  n23 <- length(intersect(B,C))
  n13 <- length(intersect(A,C))
  n123 <- length(intersect(intersect(A, B), intersect(B,C)))
  venn.plot <- draw.triple.venn(
    area1 = area1,
    area2 = area2,
    area3 = area3,
    n12 = n12,
    n23 = n23,
    n13 = n13,
    n123 = n123,
    category = cat.names,
    fill = c("orange", "forestgreen", "lightgray"),
    alpha = .6,
    cex = 2,
    cat.cex = 2,
  )

  # Add comma separators for larger numbers (https://stackoverflow.com/a/37240111/996114)
  idx <- sapply(venn.plot, function(i) grepl("text", i$name))
  for(i in 1:7){
    venn.plot[idx][[i]]$label <- format(as.numeric(venn.plot[idx][[i]]$label), big.mark=",", scientific=FALSE)
  }
}

```

```

}
venn.plot
}

png('plots/venn-rhythmic.png', w=7, h=7, u='in', res=150)
venn.rhythms <- threeway.Venn(rhythmic.east, rhythmic.west, rhythmic.merged, cat.names = c('East', 'West', 'Merged'))
grid.newpage()
grid.draw(venn.rhythms)
dev.off()

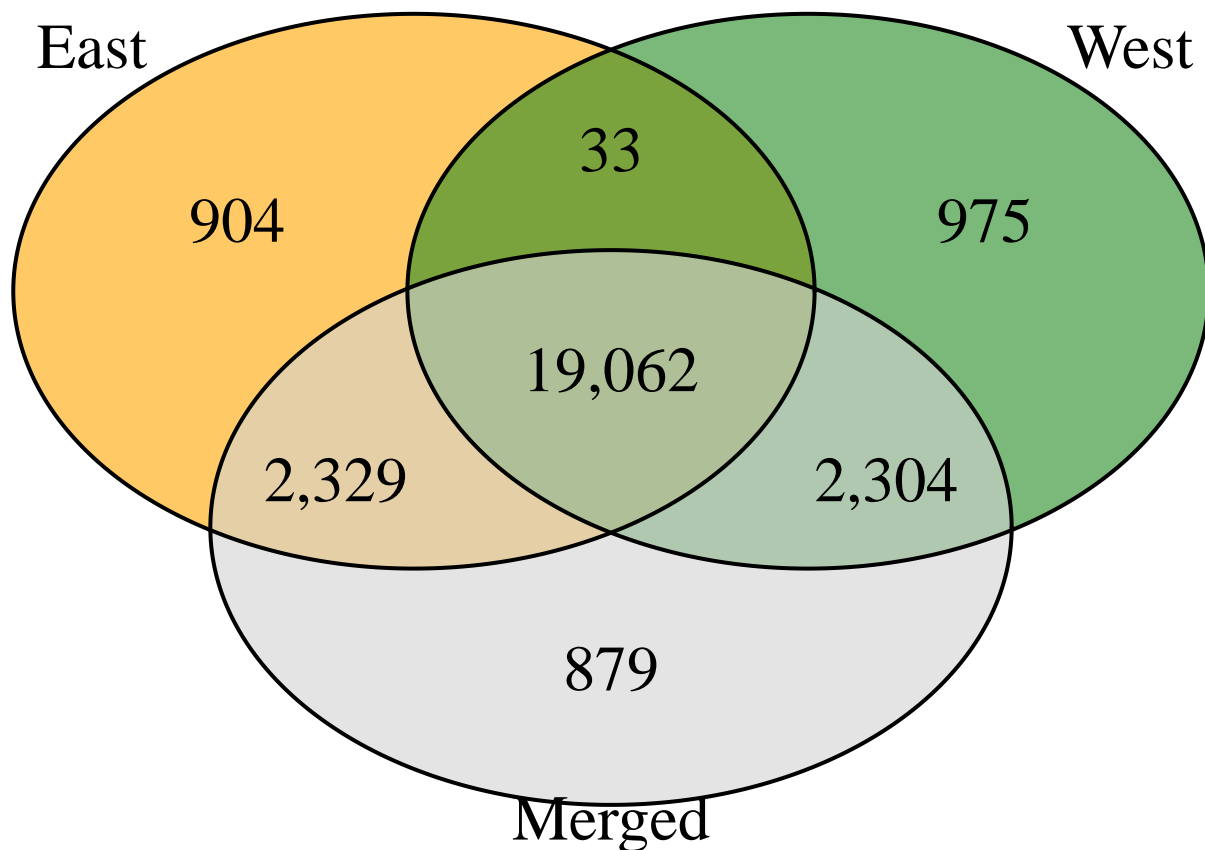
## pdf
## 2

pdf('plots/venn-rhythmic.pdf', w=7, h=7)
grid.draw(venn.rhythms)
dev.off()

## pdf
## 2

grid.newpage()
grid.draw(venn.rhythms)

```



West vs East Phase

```

cor(subset(cosopt.east, GeneID %in% rhythmic.both)$PeakPhase, subset(cosopt.west, GeneID %in% rhythmic.both)$PeakPhase)

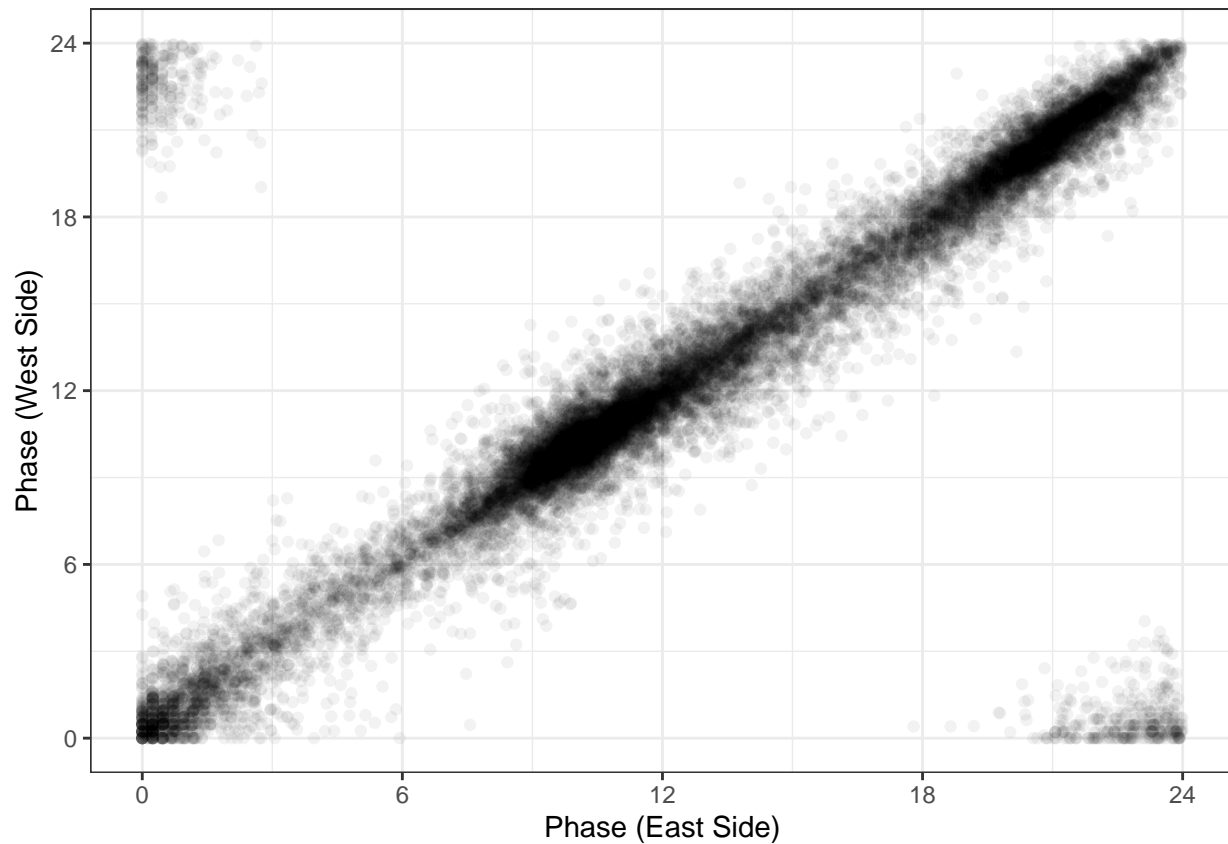
## [1] -0.004739706

```

```

cosopt.both <- subset(cosopt, GeneID %in% rhythmic.both)
ggplot(cosopt.both) +
  geom_point(aes(x = PeakPhase.E, y = PeakPhase.W), alpha=0.05) +
  scale_x_continuous(breaks=seq(0, 24, 6)) +
  scale_y_continuous(breaks=seq(0, 24, 6)) +
  xlab('Phase (East Side)') +
  ylab('Phase (West Side)') +
  theme_bw()

```



```

ggsave('plots/phases.west-vs-east.png', w=6, h=6)
ggsave('plots/phases.west-vs-east.pdf', w=6, h=6)

```

Process Data for Phase Histograms

```

cosopt.east$side <- 'East'
cosopt.west$side <- 'West'
cosopt.east.west <- rbind(cosopt.east, cosopt.west)

histogram.data <- cosopt.east.west[cosopt.east.west$GeneID %in% rhythmic.both, c('GeneID', 'PeakPhase')]
histogram.data <- subset(histogram.data, GeneID %in% rhythmic.both)
histogram.data$window <- 1
histogram.data.pre <- histogram.data
histogram.data.pre$PeakPhase <- histogram.data.pre$PeakPhase - 24
histogram.data.pre$window <- 0
histogram.data.post <- histogram.data
histogram.data.post$PeakPhase <- histogram.data.post$PeakPhase + 24
histogram.data.post$window <- 2

```

```

histogram.data.combined <- rbind(histogram.data.pre, histogram.data, histogram.data.post)

daynight <- data.frame(dawn=c(0, 24, 48, 72, 96), dusk=c(13.25 - 24, 13.25, 13.25 + 24, 13.25 + 48, 13.25 + 72, 13.25 + 96))

temperatures <- read.table('environmental-data/temp-data-table.txt', sep="\t", header=TRUE)
temperatures$ScaledTempC <- ((temperatures$TempC - min(temperatures$TempC)) * 1500) / (max(temperatures$TempC) - min(temperatures$TempC))

temperature.stats <- ddply(temperatures, .(Time), summarize, mean=mean(TempC), stderr=sqrt(var(TempC, na.rm=TRUE)))

```

```
##
```

			0%
==			4%
=====			8%
=====			12%
=====			15%
=====			19%
=====			23%
=====			27%
=====			31%
=====			35%
=====			38%
=====			42%
=====			46%
=====			50%
=====			54%
=====			58%
=====			62%
=====			65%
=====			69%
=====			73%
=====			77%

```

=====| 81%
|=====| 85%
|=====| 88%
|=====| 92%
|=====| 96%
|=====| 100%
temperature.stats.scaled <- ddp1y(temperatures, .(Time), summarize, mean=mean(ScaledTempC), stderr=sqrt
##
|
| 0%
|
|= 4%
|
|= 8%
|
|= 12%
|
|= 15%
|
|= 19%
|
|= 23%
|
|= 27%
|
|= 31%
|
|= 35%
|
|= 38%
|
|= 42%
|
|= 46%
|
|= 50%
|
|= 54%
|
|= 58%
|
|= 62%
|
|= 65%
|
|= 69%
|
|= 73%

```




temperatures

##	Time	TempC	ScaledTempC
## 1	-0.6333333	17	157.8947
## 2	-0.6333333	17	157.8947
## 3	0.3666667	15	0.0000
## 4	0.3666667	17	157.8947
## 5	1.3666667	17	157.8947
## 6	1.3666667	18	236.8421
## 7	2.3666667	18	236.8421
## 8	2.3666667	19	315.7895
## 9	3.3666667	20	394.7368
## 10	3.3666667	22	552.6316
## 11	4.3666667	23	631.5789
## 12	4.3666667	24	710.5263
## 13	5.3666667	25	789.4737
## 14	5.3666667	26	868.4211
## 15	6.3666667	28	1026.3158
## 16	6.3666667	29	1105.2632
## 17	7.3666667	29	1105.2632
## 18	7.3666667	31	1263.1579
## 19	8.3666667	31	1263.1579
## 20	8.3666667	33	1421.0526
## 21	9.3666667	32	1342.1053
## 22	9.3666667	34	1500.0000
## 23	10.3666667	32	1342.1053
## 24	10.3666667	34	1500.0000
## 25	11.3666667	32	1342.1053
## 26	11.3666667	34	1500.0000
## 27	12.3666667	29	1105.2632
## 28	12.3666667	33	1421.0526
## 29	13.3666667	27	947.3684
## 30	13.3666667	30	1184.2105
## 31	14.3666667	24	710.5263
## 32	14.3666667	26	868.4211
## 33	15.3666667	22	552.6316
## 34	15.3666667	23	631.5789
## 35	16.3666667	21	473.6842
## 36	16.3666667	21	473.6842
## 37	17.3666667	20	394.7368

```
## 38 17.3666667 21 473.6842
## 39 18.3666667 20 394.7368
## 40 18.3666667 20 394.7368
## 41 19.3666667 19 315.7895
## 42 19.3666667 19 315.7895
## 43 20.3666667 19 315.7895
## 44 20.3666667 19 315.7895
## 45 21.3666667 19 315.7895
## 46 21.3666667 18 236.8421
## 47 22.3666667 18 236.8421
## 48 22.3666667 18 236.8421
## 49 23.3666667 17 157.8947
## 50 23.3666667 17 157.8947
## 51 24.3666667 15 0.0000
## 52 24.3666667 17 157.8947
```

```
temperature.stats
```

```
##           Time mean stderr
## 1 -0.6333333 17.0 0.0
## 2 0.3666667 16.0 1.0
## 3 1.3666667 17.5 0.5
## 4 2.3666667 18.5 0.5
## 5 3.3666667 21.0 1.0
## 6 4.3666667 23.5 0.5
## 7 5.3666667 25.5 0.5
## 8 6.3666667 28.5 0.5
## 9 7.3666667 30.0 1.0
## 10 8.3666667 32.0 1.0
## 11 9.3666667 33.0 1.0
## 12 10.3666667 33.0 1.0
## 13 11.3666667 33.0 1.0
## 14 12.3666667 31.0 2.0
## 15 13.3666667 28.5 1.5
## 16 14.3666667 25.0 1.0
## 17 15.3666667 22.5 0.5
## 18 16.3666667 21.0 0.0
## 19 17.3666667 20.5 0.5
## 20 18.3666667 20.0 0.0
## 21 19.3666667 19.0 0.0
## 22 20.3666667 19.0 0.0
## 23 21.3666667 18.5 0.5
## 24 22.3666667 18.0 0.0
## 25 23.3666667 17.0 0.0
## 26 24.3666667 16.0 1.0
```

Plot Phase Histograms

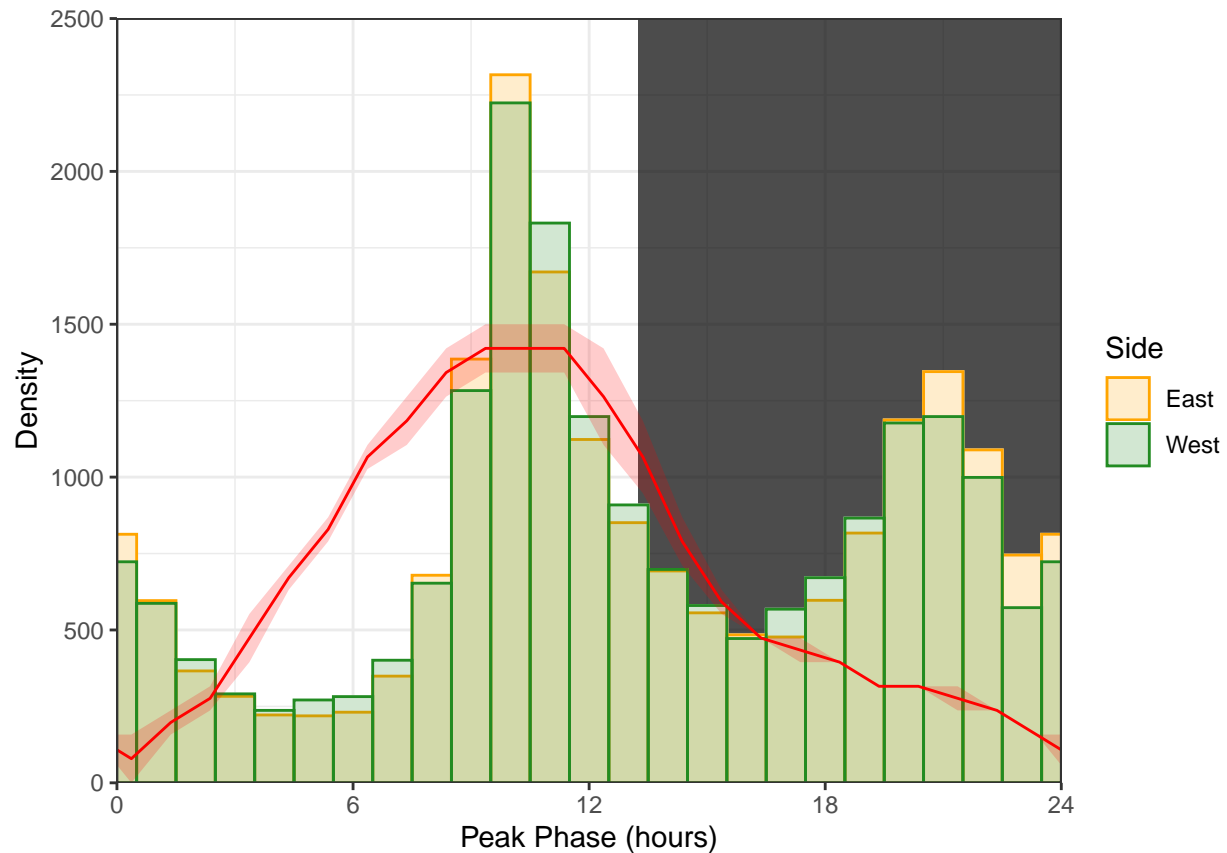
```
p <- ggplot() +
  geom_rect(data=daynight, aes(xmin=dawn, xmax=dusk), fill='black', ymin=-10000, ymax=10000, alpha=0.7) +
  geom_histogram(data=subset(histogram.data.combined, side=='West'), aes(x=PeakPhase, y=..count..), color='red', fill='red', alpha=0.2) +
  geom_histogram(data=subset(histogram.data.combined, side=='East'), aes(x=PeakPhase, y=..count..), color='blue', fill='blue', alpha=0.2) +
  geom_histogram(data=histogram.data.combined, aes(x=PeakPhase, color=side, fill=side, y=..count..), alpha=0.2) +
  geom_ribbon(data=temperature.stats.scaled, aes(x=Time, ymin=min, ymax=max), fill='red', alpha=0.2) +
  geom_line(data=temperature.stats.scaled, aes(x=Time, y=mean), color='red') +
  labs(x = 'Peak Phase (hours)', y = 'Density') +
```

```

scale_color_manual(name = 'Side', values = c(east.color, west.color)) +
scale_fill_manual(name = 'Side', values = c(east.color, west.color)) +
scale_x_continuous(breaks=seq(0, 24, 6)) +
coord_cartesian(xlim=c(0, 24), ylim=c(0, 2500), expand=F) +
theme_bw()

```

p



```

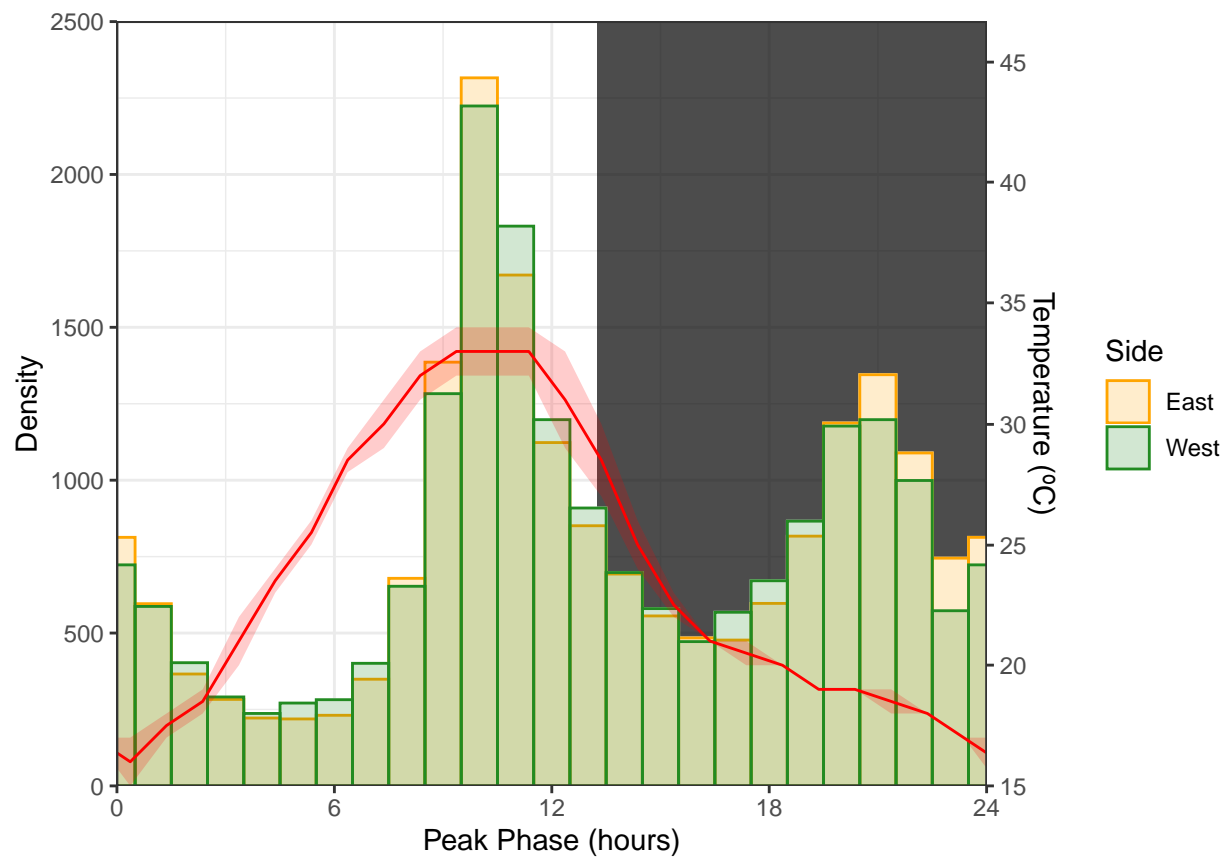
ggsave('plots/phase-histogram.temperature.png', w=6, h=5)
ggsave('plots/phase-histogram.temperature.pdf', w=6, h=5)

```

```

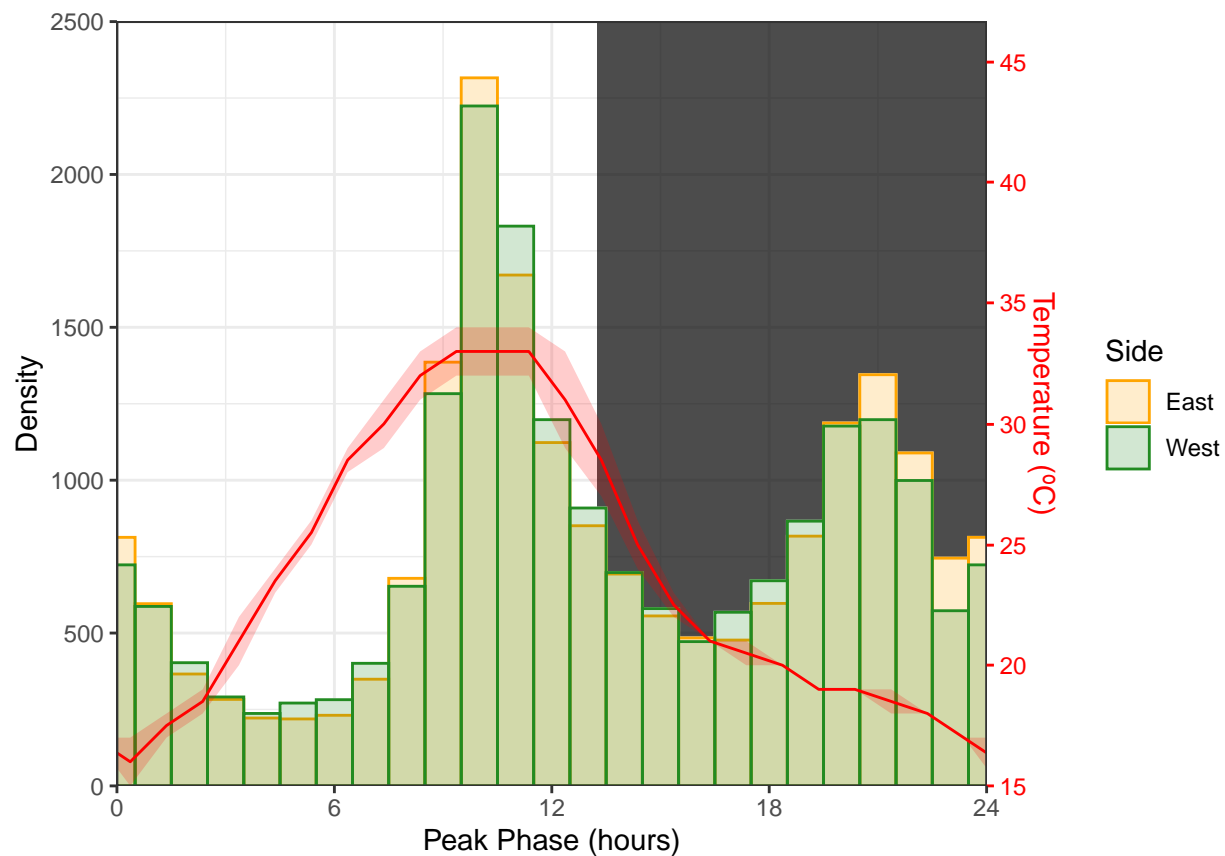
scale_m <- (max(temperatures$TempC) - min(temperatures$TempC)) / (1500 - p$coordinates$limits$y[1])
scale_b <- min(temperatures$TempC)
scale_temp_max <- p$coordinates$limits$y[2] * scale_m + scale_b
scale_temp_min <- min(temperatures$TempC)
p + scale_y_continuous(sec.axis = sec_axis(~.*scale_m + scale_b, name = "Temperature (°C)", breaks=seq(

```



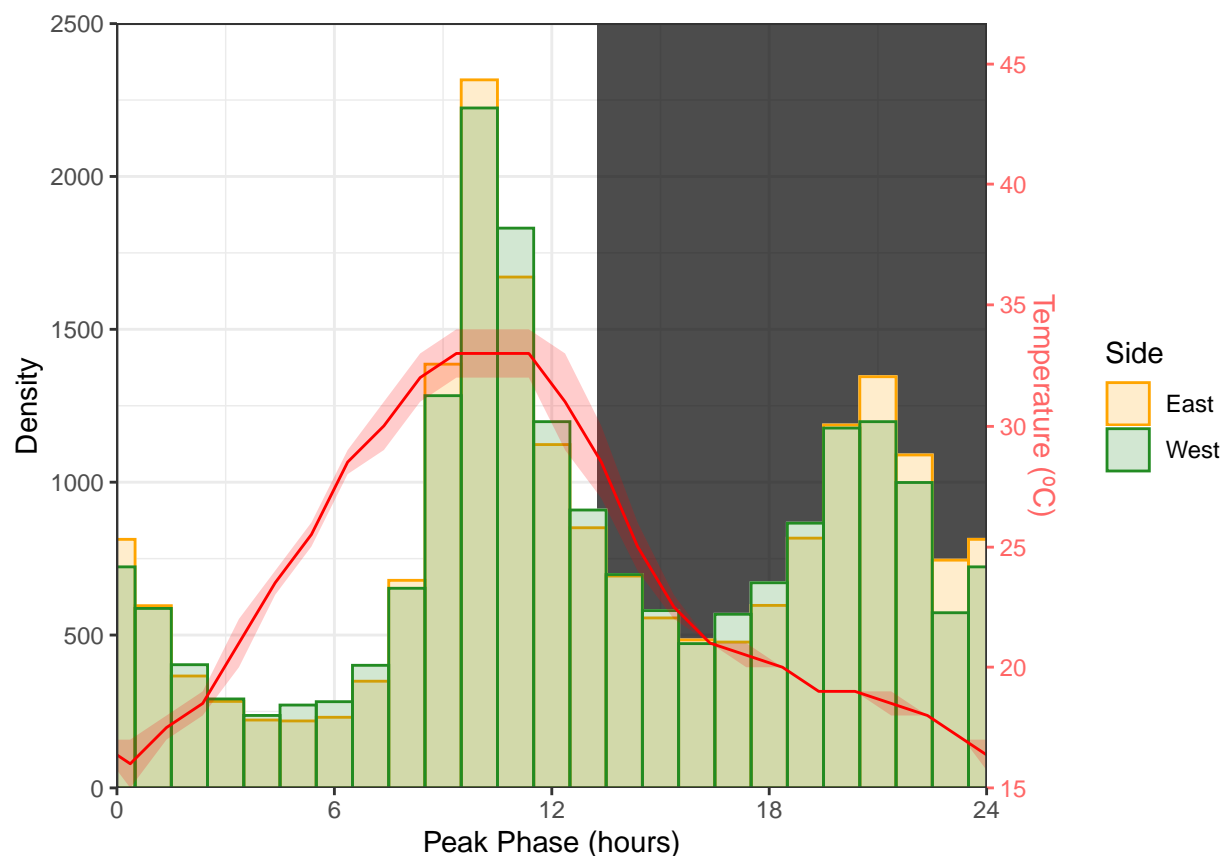
```
ggsave('plots/phase-histogram.temperature-axis.png', w=6, h=5)
ggsave('plots/phase-histogram.temperature-axis.pdf', w=6, h=5)
```

```
p + scale_y_continuous(sec.axis = sec_axis(~.*scale_m + scale_b, name = "Temperature (°C)", breaks=seq(
  theme(
    axis.title.y.right = element_text(color = "red"),
    axis.text.y.right = element_text(color = "red"),
    axis.ticks.y.right = element_line(color = "red"),
  )
)
```



```
ggsave('plots/phase-histogram.temperature-axis-red.png', w=6, h=5)
ggsave('plots/phase-histogram.temperature-axis-red.pdf', w=6, h=5)

p + scale_y_continuous(sec.axis = sec_axis(~.*scale_m + scale_b, name = "Temperature (°C)", breaks=seq(
  theme(
    axis.title.y.right = element_text(color = alpha("red", 0.6)),
    axis.text.y.right = element_text(color = alpha("red", 0.6)),
    axis.ticks.y.right = element_line(color = alpha("red", 0.6)),
  )
)
```



```
ggsave('plots/phase-histogram.temperature-axis-lightred.png', w=6, h=5)
ggsave('plots/phase-histogram.temperature-axis-lightred.pdf', w=6, h=5)
```

The cosopt-processed.txt file that we just generated should have an MD5 checksum of 2fda73974466f805a22b1941b3f958f

```
md5sum(cosopt.processed.file)
```

```
## cosopt-processed/cosopt-processed.txt
## "2fda73974466f805a22b1941b3f958fe"
```

Plot Amplitude Differences Summary

```
plot.ampldiff.summary <- function() {
  timecourse.w <- subset(timecourse, gene %in% west.high)
  timecourse.e <- subset(timecourse, gene %in% east.high)

  timecourse.w <- merge(timecourse.w, cosopt[, c('GeneID', 'MeanExpLev')], by.x='gene', by.y='GeneID')
  timecourse.e <- merge(timecourse.e, cosopt[, c('GeneID', 'MeanExpLev')], by.x='gene', by.y='GeneID')

  timecourse.w$mean.norm <- timecourse.w$mean / timecourse.w$MeanExpLev
  timecourse.e$mean.norm <- timecourse.e$mean / timecourse.e$MeanExpLev

  timecourse.w <- dcast(timecourse.w, hour ~ side, mean, value.var='mean.norm')
  timecourse.e <- dcast(timecourse.e, hour ~ side, mean, value.var='mean.norm')

  timecourse.w <- melt(timecourse.w, id.vars='hour', variable.name='side', value.name='mean.norm', na.rm=TRUE)
  timecourse.e <- melt(timecourse.e, id.vars='hour', variable.name='side', value.name='mean.norm', na.rm=TRUE)
```

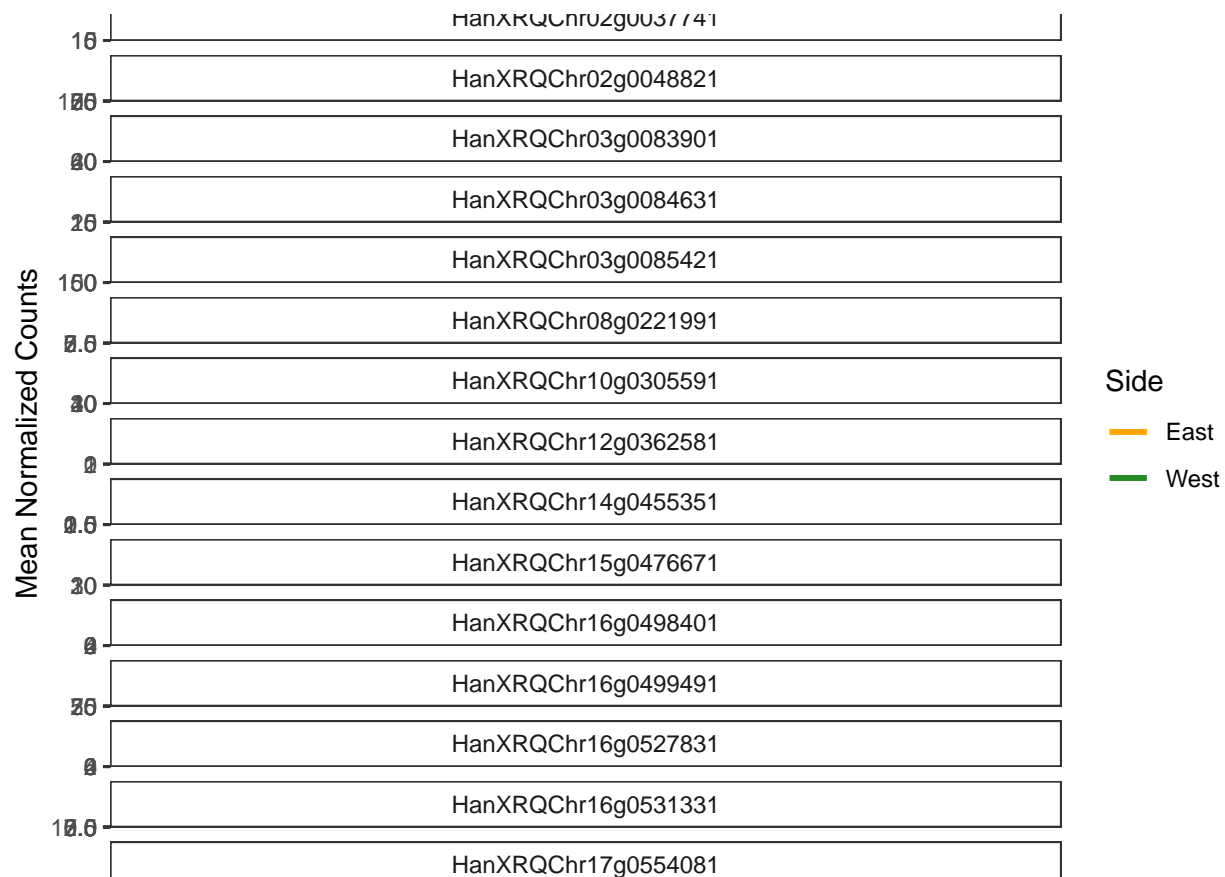
```

timecourse.w$high.side <- paste0('West Higher (n=', length(west.high), ")")
timecourse.e$high.side <- paste0('East Higher (n=', length(east.high), ")")
timecourse.we <- rbind(timecourse.w, timecourse.e)

p <- ggplot(timecourse.we, aes(x=hour, y=mean.norm, color=side)) +
  geom_line(size=1) +
  labs(x = 'Time (hours)', y = 'Mean of (Mean Normalized Counts / Mean Expression Level)') +
  scale_x_continuous(breaks=seq(0, 48, 12)) +
  scale_color_manual(name = 'Orientation', values = c(east.color, west.color)) +
  facet_wrap(~ high.side, ncol=1, scales='free_y')
print(p)
p
}

expdiff <- subset(cosopt, GeneID %in% rhythmic.both & abs(exp.diff.log2) > 0.6 & (MeanExpLev.W > 0.5 | 
plot.timecourse(expdiff$GeneID, lights.off = 13.25)

```



```

ggsave(paste0('plots/exp-diff.png'), w=6, h=25)
ggsave(paste0('plots/exp-diff.pdf'), w=6, h=25)
write.table(expdiff, 'cosopt-processed/cosopt-processed.exp-diff.txt', sep = "\t", quote = FALSE, col.names=NA)

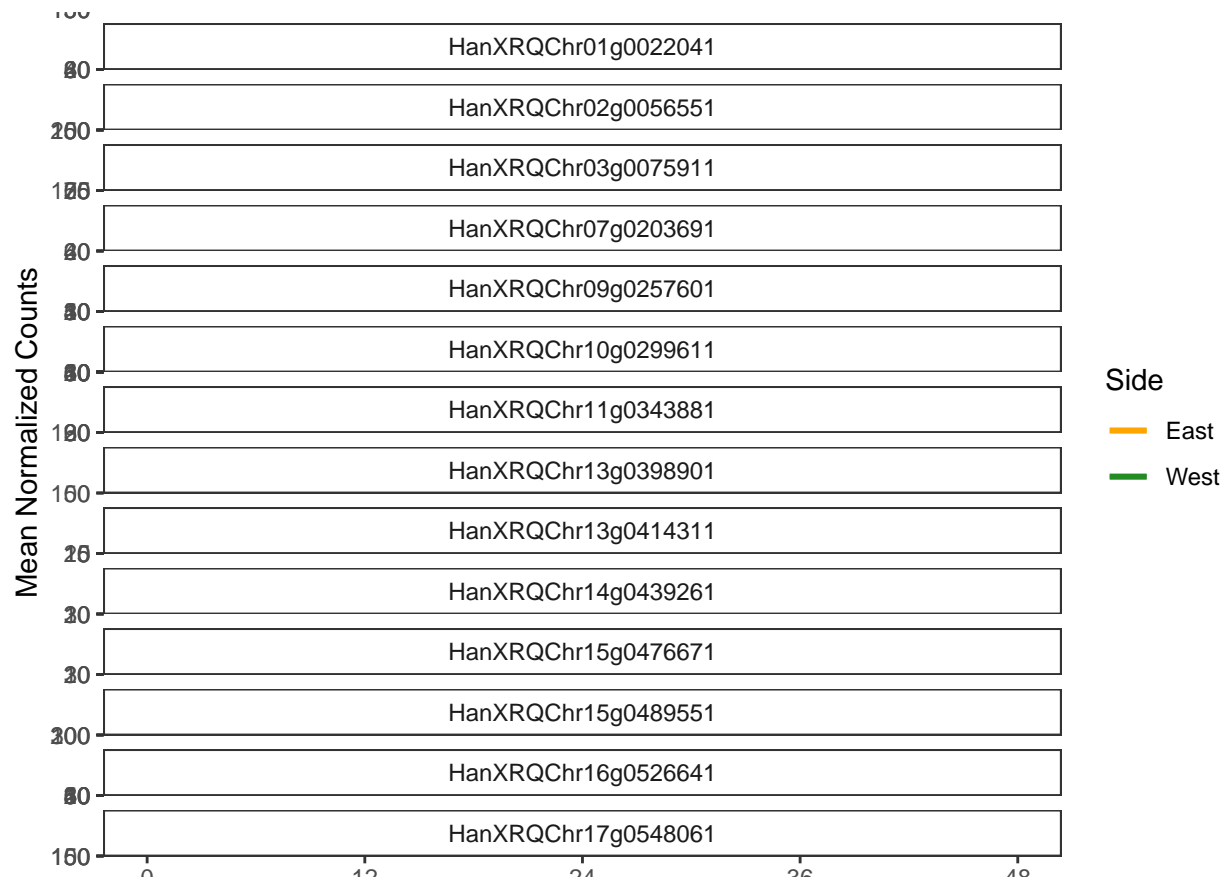
exp <- rownames(expdiff)
exp.e <- subset(cosopt, GeneID %in% exp & exp.diff.log2 < 0)$GeneID
exp.w <- subset(cosopt, GeneID %in% exp & exp.diff.log2 > 0)$GeneID

```

```
ampdiff <- subset(cosopt, GeneID %in% rhythmic.both & abs(amp.diff) > 0.25 & (MeanExpLev.E > 10 | MeanE
amp <- rownames(ampdiff)

amp.e <- subset(cosopt, GeneID %in% amp & amp.diff < 0)$GeneID
amp.w <- subset(cosopt, GeneID %in% amp & amp.diff > 0)$GeneID

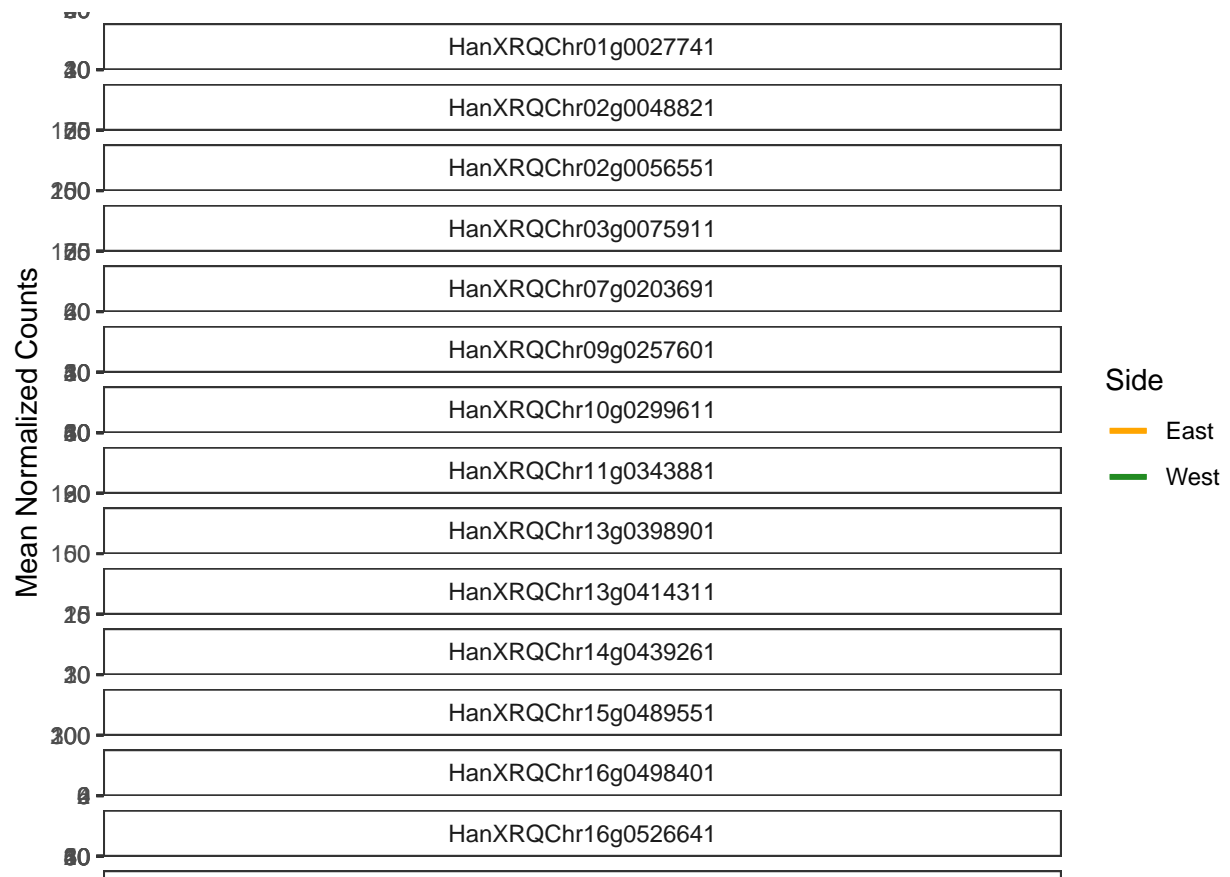
plot.timecourse(amp, lights.off = 13.25)
```



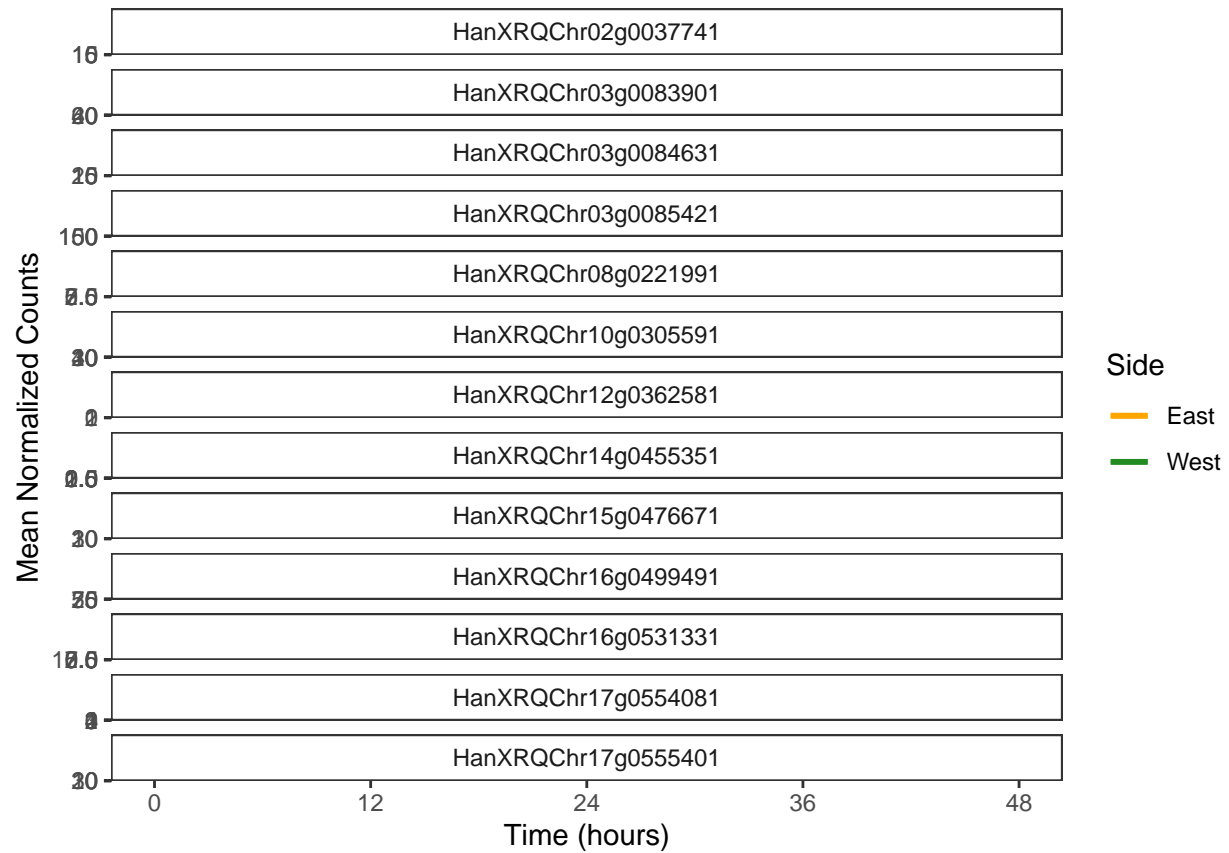
```
ggsave(paste0('plots/amp-diff.png'), w=6, h=23)
ggsave(paste0('plots/amp-diff.pdf'), w=6, h=23)
write.table(ampdiff, 'cosopt-processed/cosopt-processed.amp-diff.txt', sep = "\t", quote = FALSE, col.n

west.high <- union(exp.w, amp.w)
east.high <- union(exp.e, amp.e)

plot.timecourse(west.high, lights.off = 13.25)
```

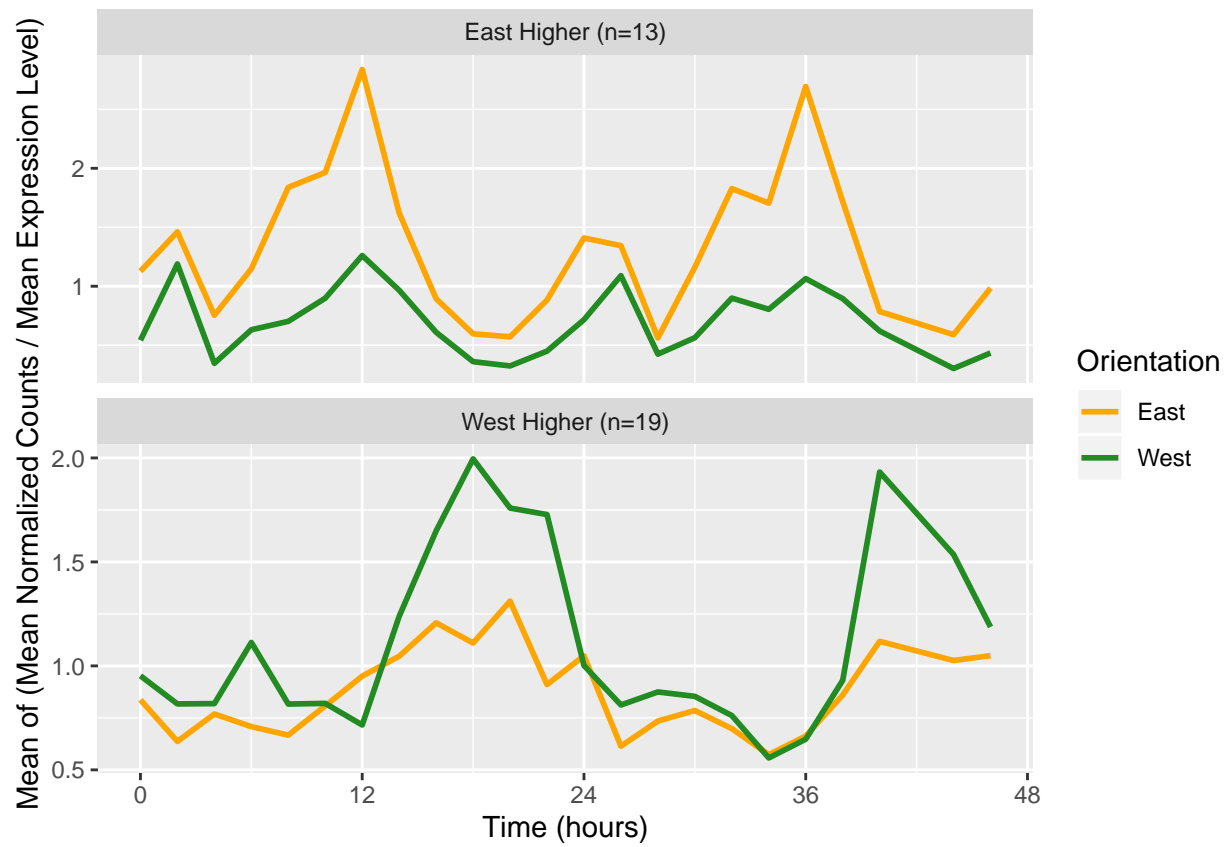



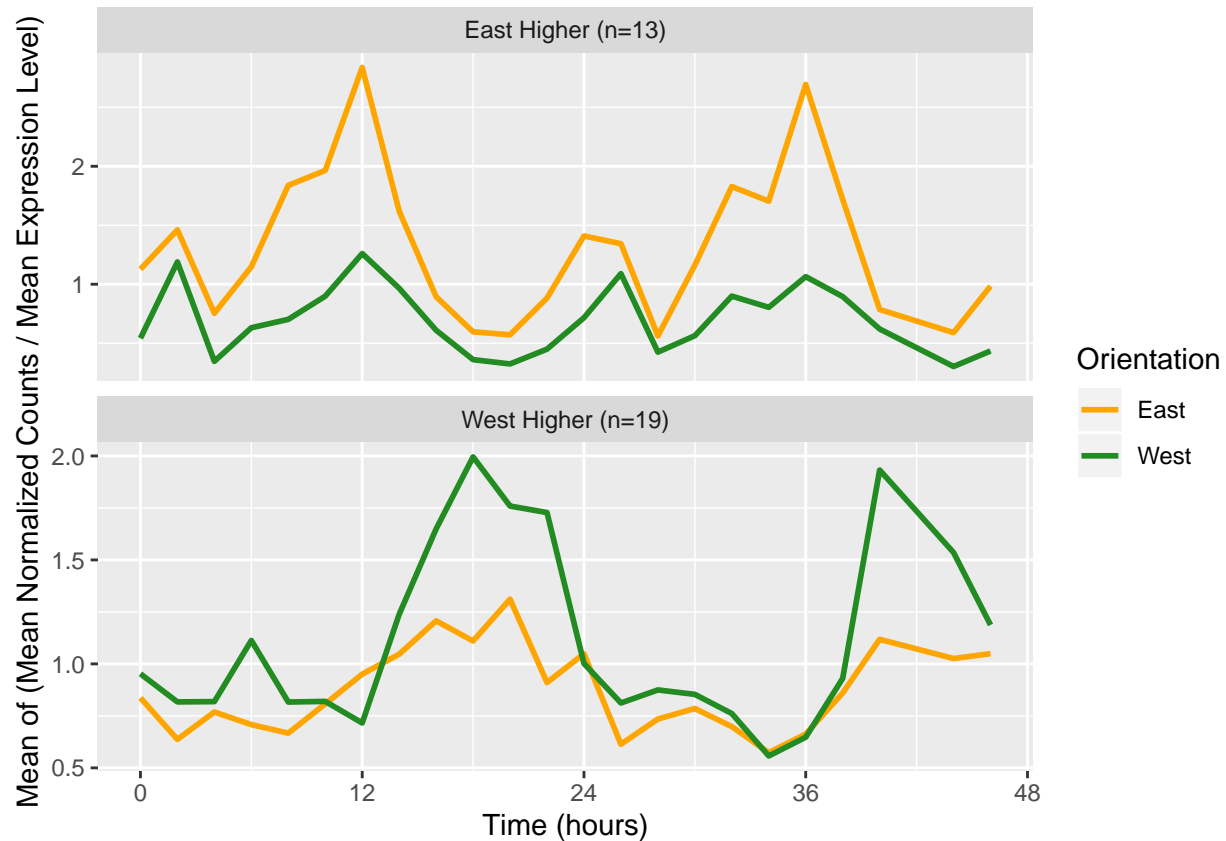
```
ggsave('plots/amp-exp-diff.west-high.png', w=6, h=30)
ggsave('plots/amp-exp-diff.west-high.pdf', w=6, h=30)
plot.timecourse(east.high, lights.off = 13.25)
```



```
ggsave('plots/amp-exp-diff.east-high.png', w=6, h=21)
ggsave('plots/amp-exp-diff.east-high.pdf', w=6, h=21)

plot.ampdiff.summary()
```



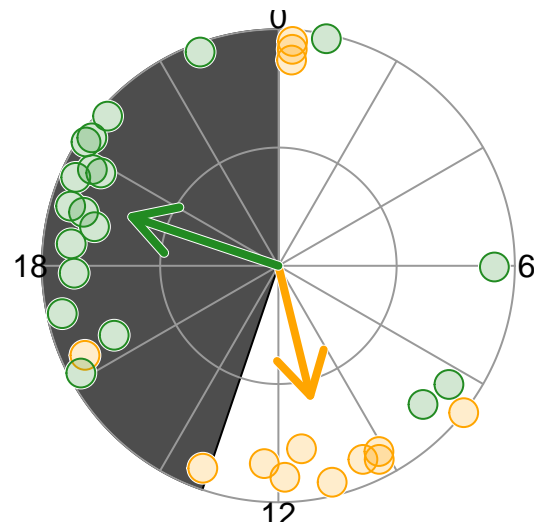
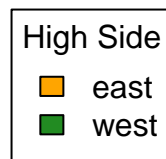


```
# ggsave("plots/amp-exp-diff-summary.png", w=5, h=7)
write.table(subset(cosopt, GeneID %in% west.high), 'cosopt-processed/cosopt-processed.amp-exp-diff.west.txt')
write.table(subset(cosopt, GeneID %in% east.high), 'cosopt-processed/cosopt-processed.amp-exp-diff.east.txt')

# Polar
east.high.phase <- subset(cosopt, GeneID %in% east.high)$PeakPhase.E
west.high.phase <- subset(cosopt, GeneID %in% west.high)$PeakPhase.W

radius <- rep(1, length(east.high.phase) + length(west.high.phase))
phases <- c(east.high.phase, west.high.phase)
groups <- factor(c(rep('east', length(east.high.phase)), rep('west', length(west.high.phase))))
set.seed(1949); noise <- rnorm(length(radius), 0, 0.05)

polar.plot(radius + noise - max(noise), phases, pch=21, grp=groups, col=c(east.color, west.color), hour=0:48)
```



```
png('plots/amp-exp-diff.png', w=7, h=7, u='in', res=150)
polar.plot(radius + noise - max(noise), phases, pch=21, grp=groups, col=c(east.color, west.color), hour=
dev.off()
```

```
## pdf
## 2
```

```
pdf('plots/amp-exp-diff.pdf', w=7, h=7)
polar.plot(radius + noise - max(noise), phases, pch=21, grp=groups, col=c(east.color, west.color), hour=
dev.off()
```

```
## pdf
## 2
```

Asymmetric Rhythm Polar Plot

```
asym.rhythm <- function(side, p1=0.01, p2=0.1, .cosopt=csopt, amp.min=0, exp.min=0, per.buffer=0, per.
  if (side == 'east') {
    return(subset(.cosopt, pMMC.Beta.E < p1 & (is.na(pMMC.Beta.W) | pMMC.Beta.W >= p2) & RelAmp.E >= amp
  } else if (side == 'west') {
    return(subset(.cosopt, pMMC.Beta.W < p1 & (is.na(pMMC.Beta.E) | pMMC.Beta.E >= p2) & RelAmp.W >= amp
  } else {
    print("Need to provide a valid value for side: 'east' or 'west'.")
  }
}
```

```
east.rhythmic <- rownames(asym.rhythm(s='east', p1=0.001, p2=0.1, amp.min=amp.min, exp.min=exp.min, per
west.rhythmic <- rownames(asym.rhythm(s='west', p1=0.001, p2=0.1, amp.min=amp.min, exp.min=exp.min, per
```

```

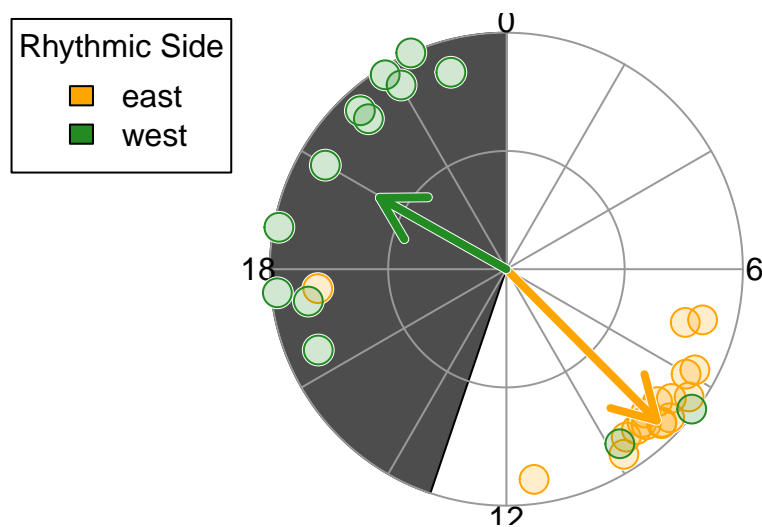
east.phase <- subset(cosopt, GeneID %in% east.rhythmic)$PeakPhase.E
west.phase <- subset(cosopt, GeneID %in% west.rhythmic)$PeakPhase.W

write.table(subset(cosopt, GeneID %in% east.rhythmic), 'cosopt-processed/cosopt-processed.asymmetric-rhythmic-east.txt',
write.table(subset(cosopt, GeneID %in% west.rhythmic), 'cosopt-processed/cosopt-processed.asymmetric-rhythmic-west.txt',

radius <- rep(1, length(east.phase) + length(west.phase))
phases <- c(east.phase, west.phase)
groups <- factor(c(rep('east', length(east.phase)), rep('west', length(west.phase))))
set.seed(0709); noise <- rnorm(length(radius), 0, 0.05)

polar.plot(radius + noise - max(noise), phases, pch=21, grp=groups, col=c(east.color, west.color), hour=0:23)

```



```

png('plots/asymmetric-rhythms.png', w=7, h=7, u='in', res=150)
polar.plot(radius + noise - max(noise), phases, pch=21, grp=groups, col=c(east.color, west.color), hour=0:23)
dev.off()

```

```

## pdf
## 2

```

```

pdf('plots/asymmetric-rhythms.pdf', w=7, h=7)
polar.plot(radius + noise - max(noise), phases, pch=21, grp=groups, col=c(east.color, west.color), hour=0:23)
dev.off()

```

```

## pdf

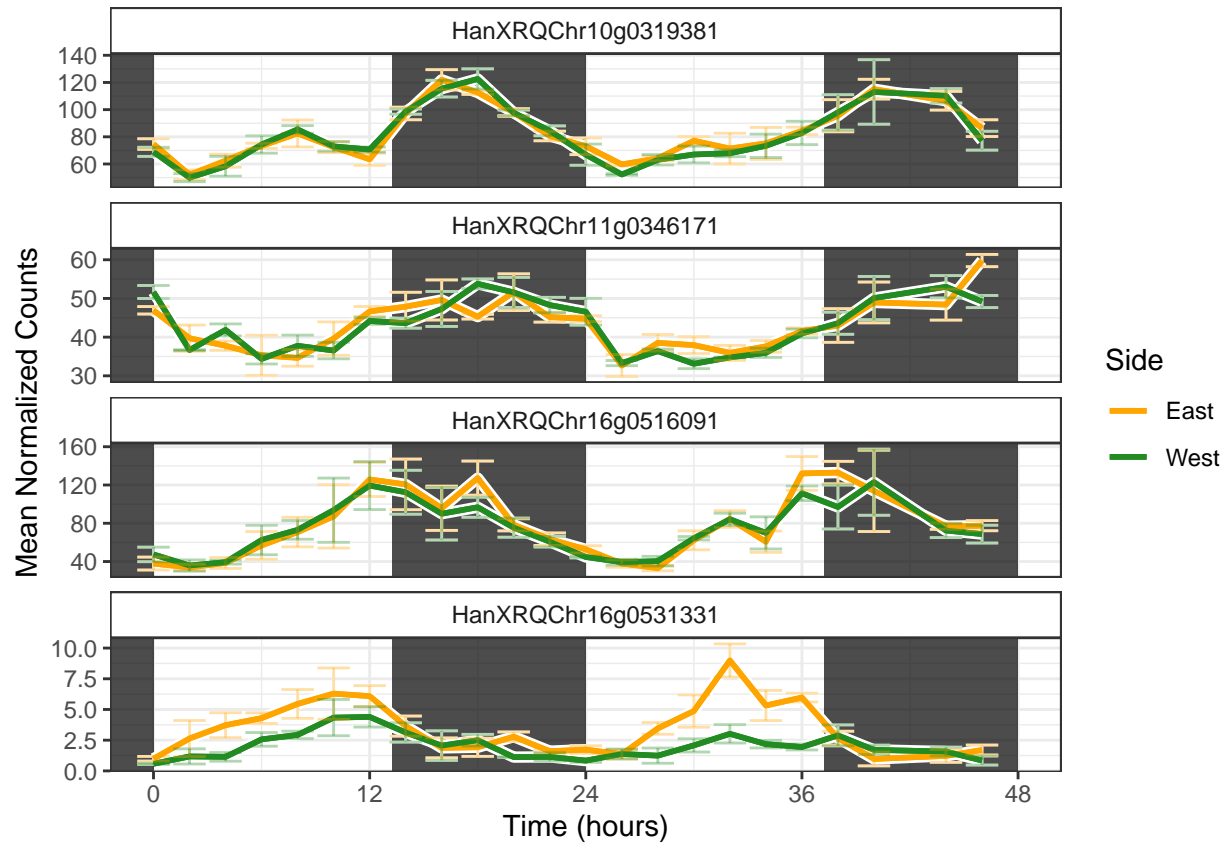
```

```
## 2
```

Plotting GWAS Candidates

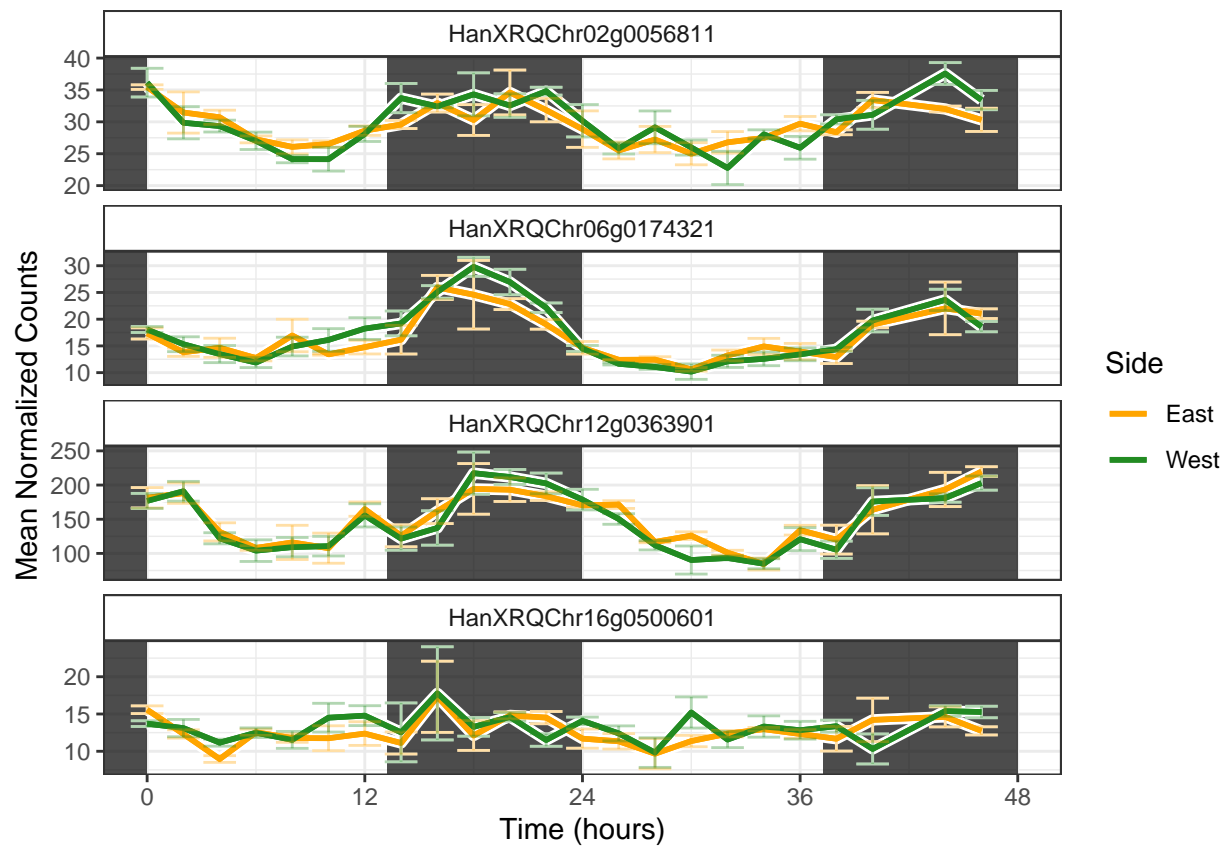
```
onset.time <- c('HanXRQChr10g0319381', 'HanXRQChr16g0516091', 'HanXRQChr16g0531331', 'HanXRQChr11g0346171')
nocturnal.reorientation <- c('HanXRQChr02g0056811', 'HanXRQChr16g0500601', 'HanXRQChr12g0363901', 'HanXRQChr16g0500601')
shoot.movement.pc1 <- c('HanXRQChr08g0210081', 'HanXRQChr03g0091141', 'HanXRQChr10g0308851')

plot.timecourse(onset.time, lights.off=13.25)
```



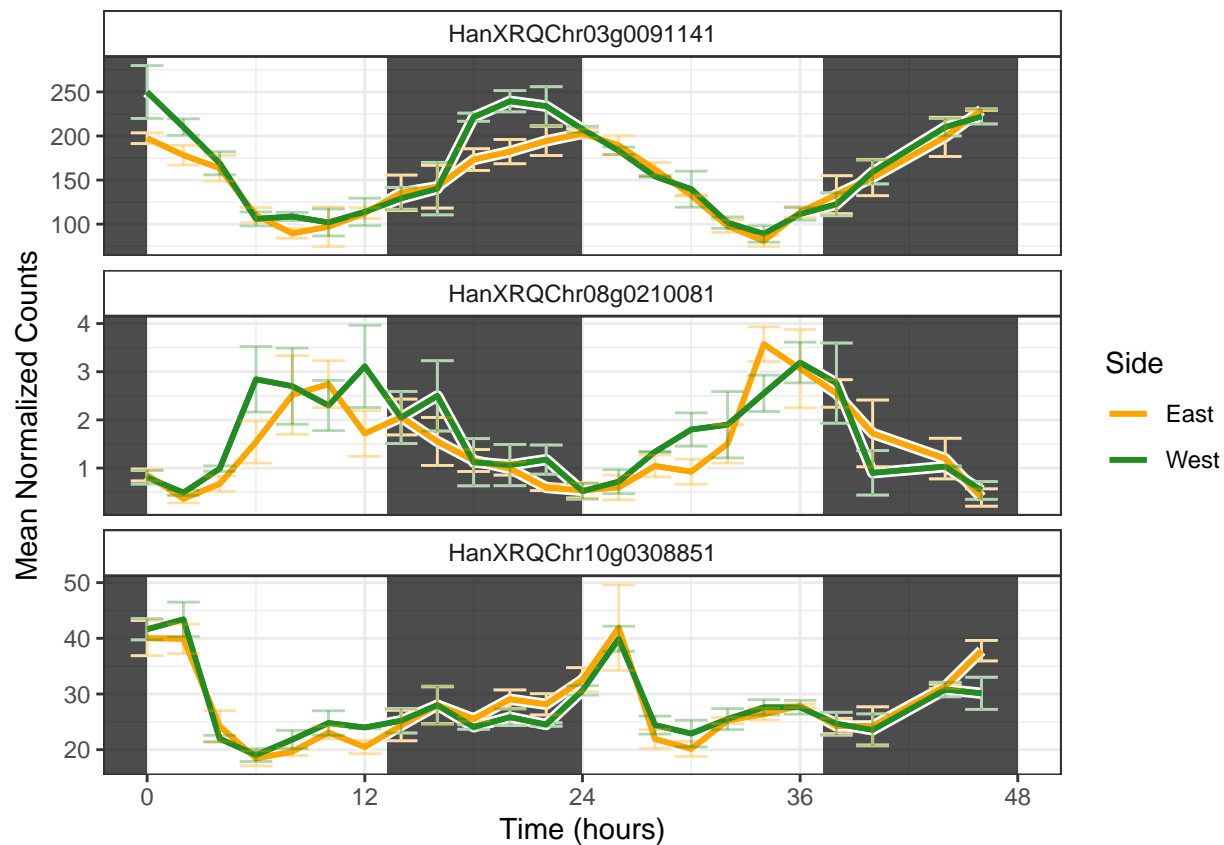
```
ggsave('plots/gwas.onset-time.png', w=4, h=6)
ggsave('plots/gwas.onset-time.pdf', w=4, h=6)

plot.timecourse(nocturnal.reorientation, lights.off=13.25)
```



```
ggsave('plots/gwas.nocturnal-reorientation.png', w=4, h=6)
ggsave('plots/gwas.nocturnal-reorientation.pdf', w=4, h=6)

plot.timecourse(shoot.movement.pc1, lights.off=13.25)
```

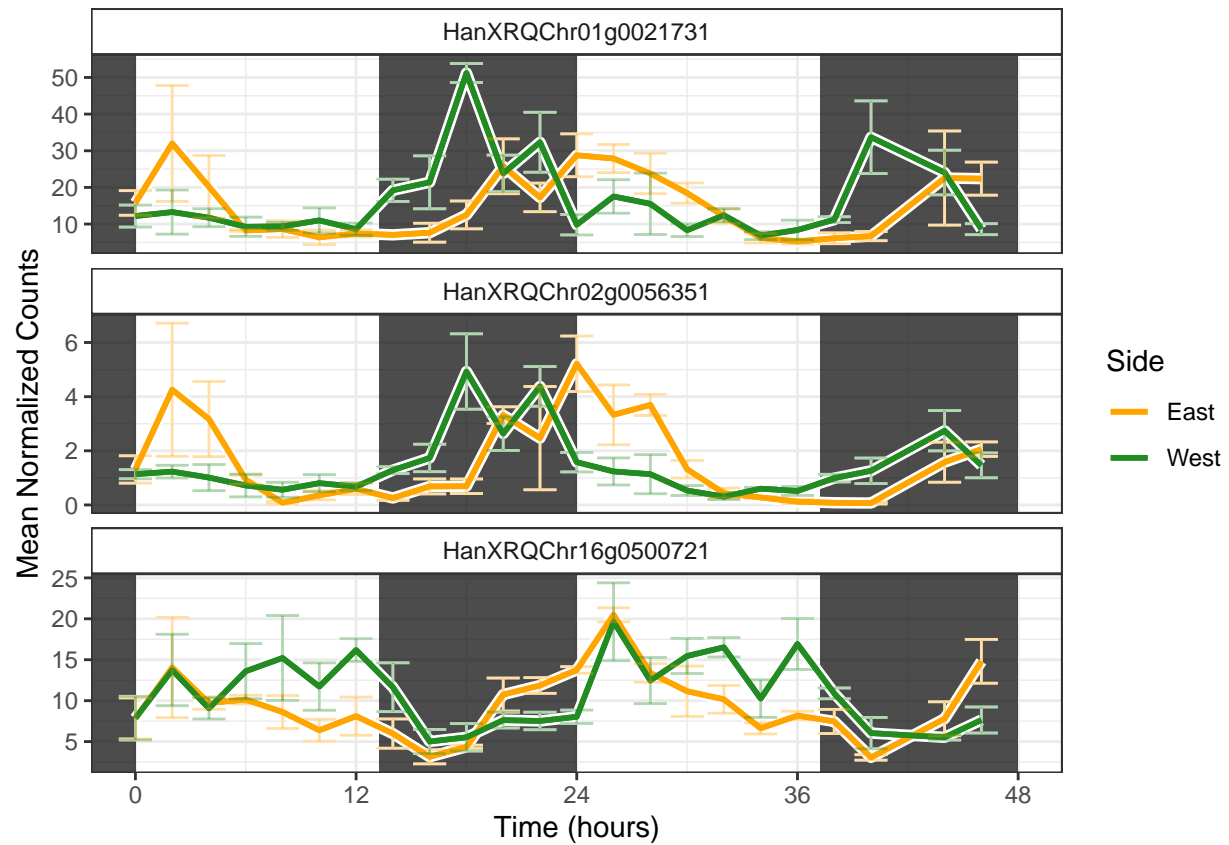
```
ggsave('plots/gwas.shoot-movement-pc1.png', w=4, h=4.7)
ggsave('plots/gwas.shoot-movement-pc1.pdf', w=4, h=4.7)
```

```
# Three genes implicated in Auxin- and Gibberillin-mediated growth are phase shifted between East and West
# HanXRQChr01g0021731 AT2G01420 PIN4 Auxin efflux carrier family protein
# HanXRQChr02g0056351 AT3G28857 PRE5: PACLOBUTRAZOL RESISTANCE 5 basic helix-loop-helix (bHLH) DNA-binding protein
# HanXRQChr16g0500721 AT3G04730 IAA16 indoleacetic acid-induced protein 16
```

```
# This one has a pMMC-Beta value of 0.05225100 for the East side and just misses the cutoff of 0.05.
# HanXRQChr13g0402621 AT4G38840 SAUR-like auxin-responsive protein family (According to https://academic.oup.com/pcp/article/34/1/1/1)
```

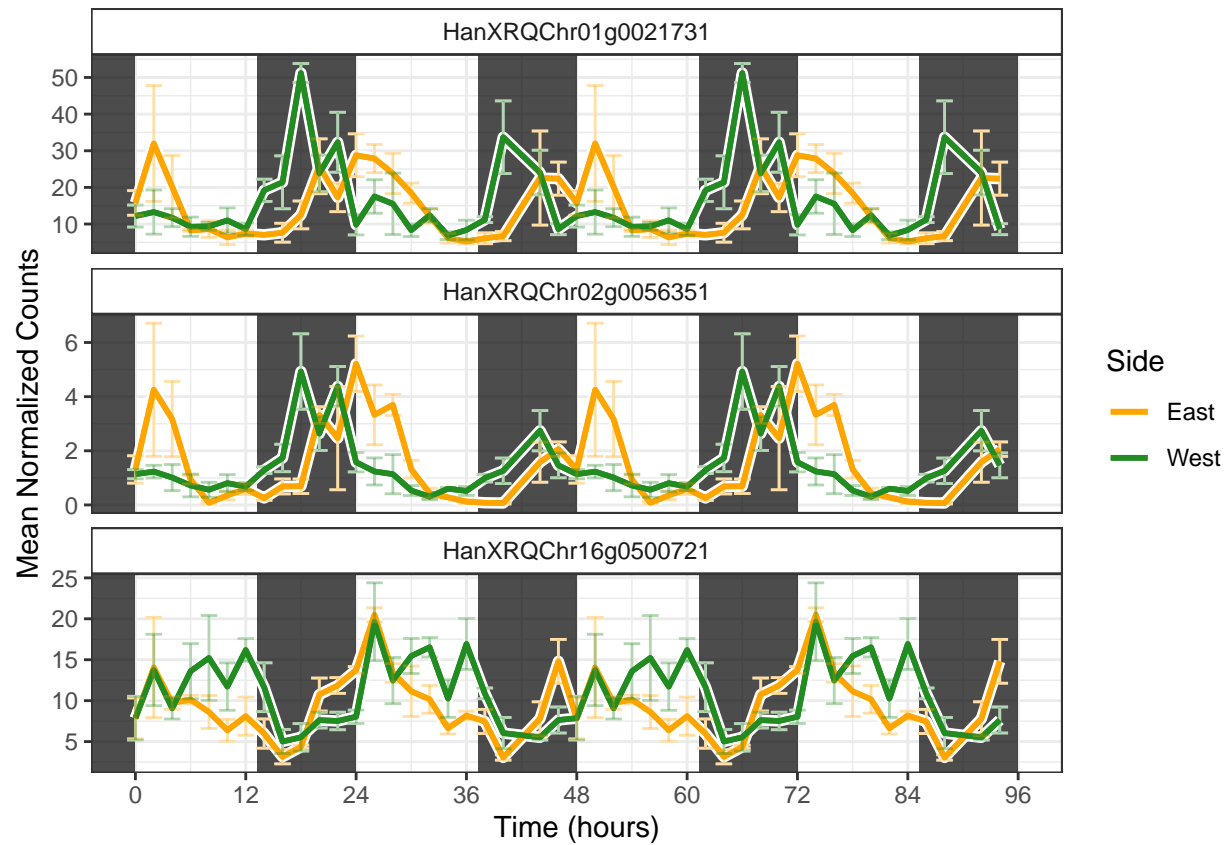
```
phase.shifted.genes <- c('HanXRQChr01g0021731', 'HanXRQChr02g0056351', 'HanXRQChr16g0500721')
```

```
plot.timecourse(phase.shifted.genes, lights.off = 13.25)
```



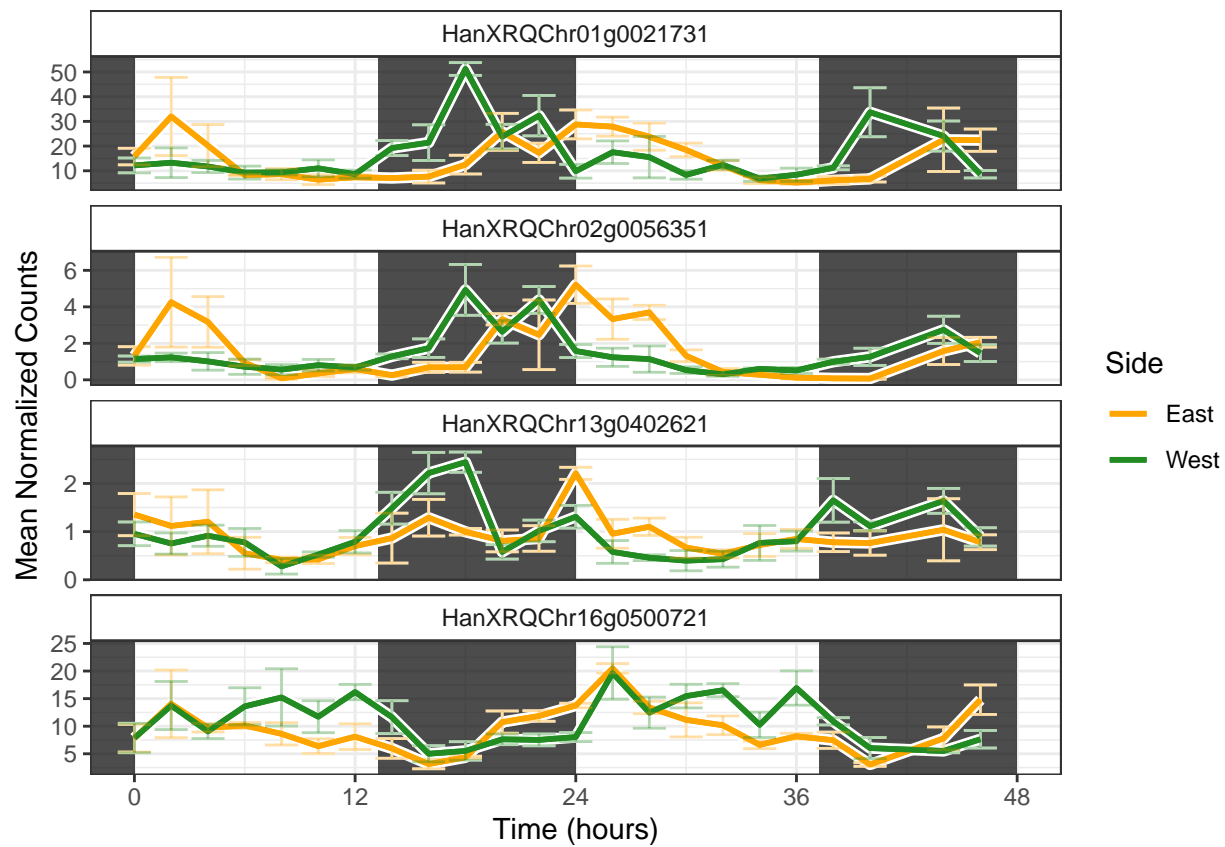
```
ggsave('plots/phase-shifted.png', w=4, h=4.7)
ggsave('plots/phase-shifted.pdf', w=4, h=4.7)

plot.timecourse(phase.shifted.genes, lights.off = 13.25, double.plot = TRUE)
```



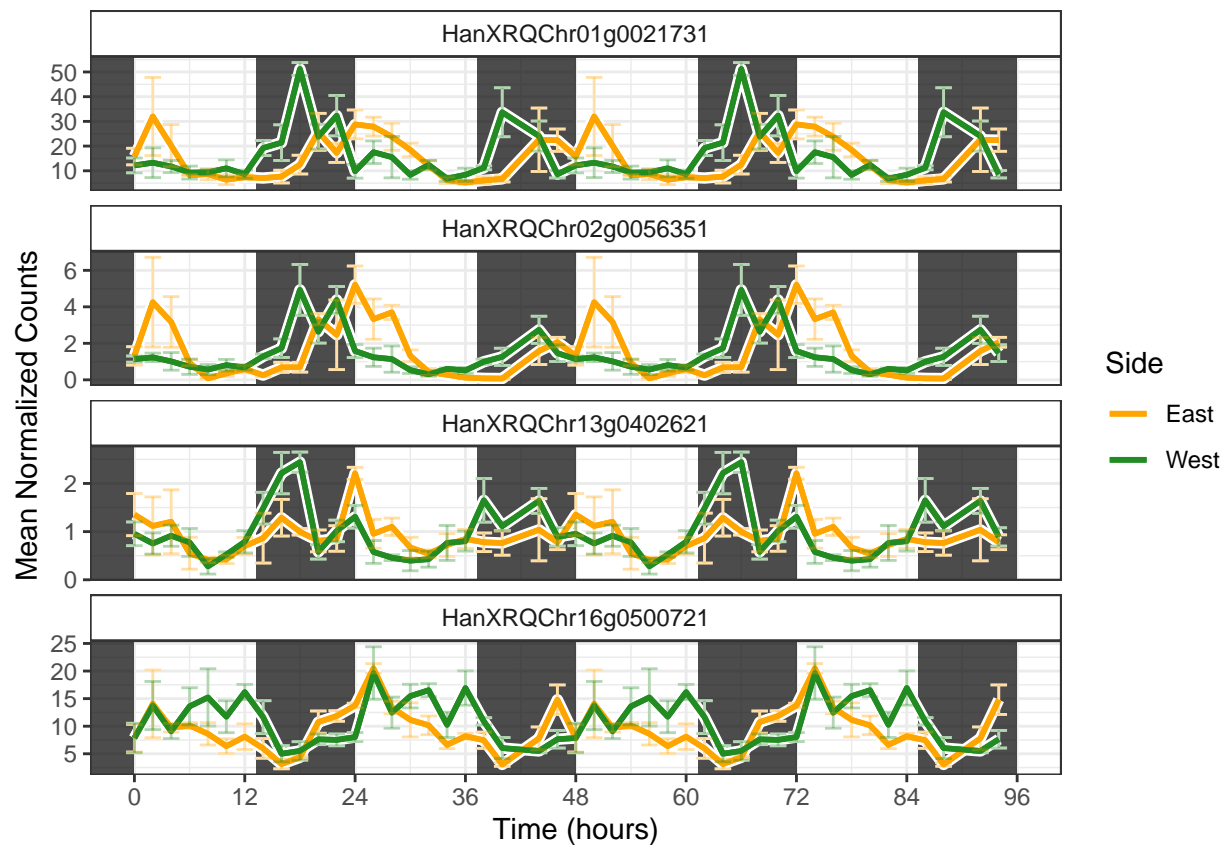
```
ggsave('plots/phase-shifted.double-plotted.png', w=6.5, h=4.7)
ggsave('plots/phase-shifted.double-plotted.pdf', w=6.5, h=4.7)

plot.timecourse(c(phase.shifted.genes, 'HanXRQChr13g0402621'), lights.off = 13.25)
```



```
ggsave('plots/phase-shifted.with-SAUR14.png', w=4, h=6)
ggsave('plots/phase-shifted.with-SAUR14.pdf', w=4, h=6)

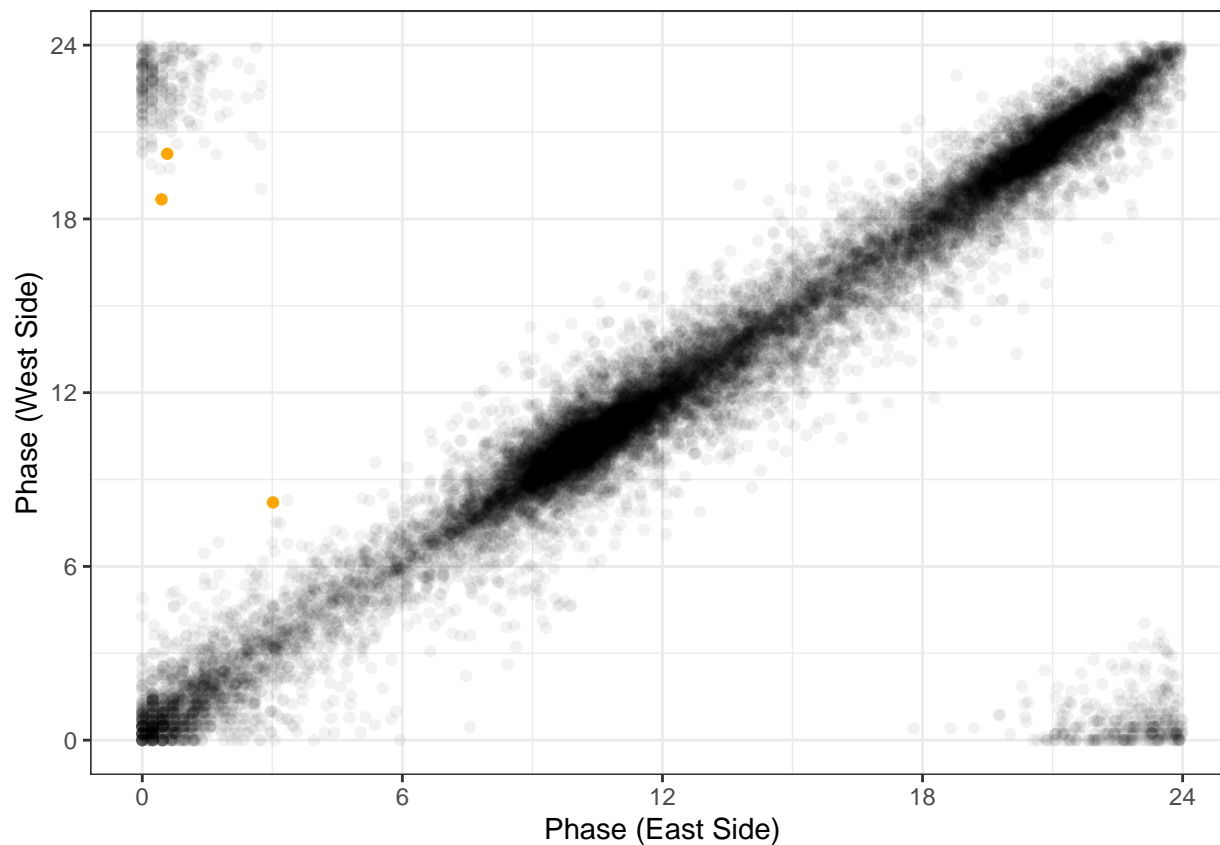
plot.timecourse(c(phase.shifted.genes, 'HanXRQChr13g0402621'), lights.off = 13.25, double.plot = TRUE)
```



```
ggsave('plots/phase-shifted.double-plotted.with-SAUR14.png', w=6.5, h=6)
ggsave('plots/phase-shifted.double-plotted.with-SAUR14.pdf', w=6.5, h=6)
```

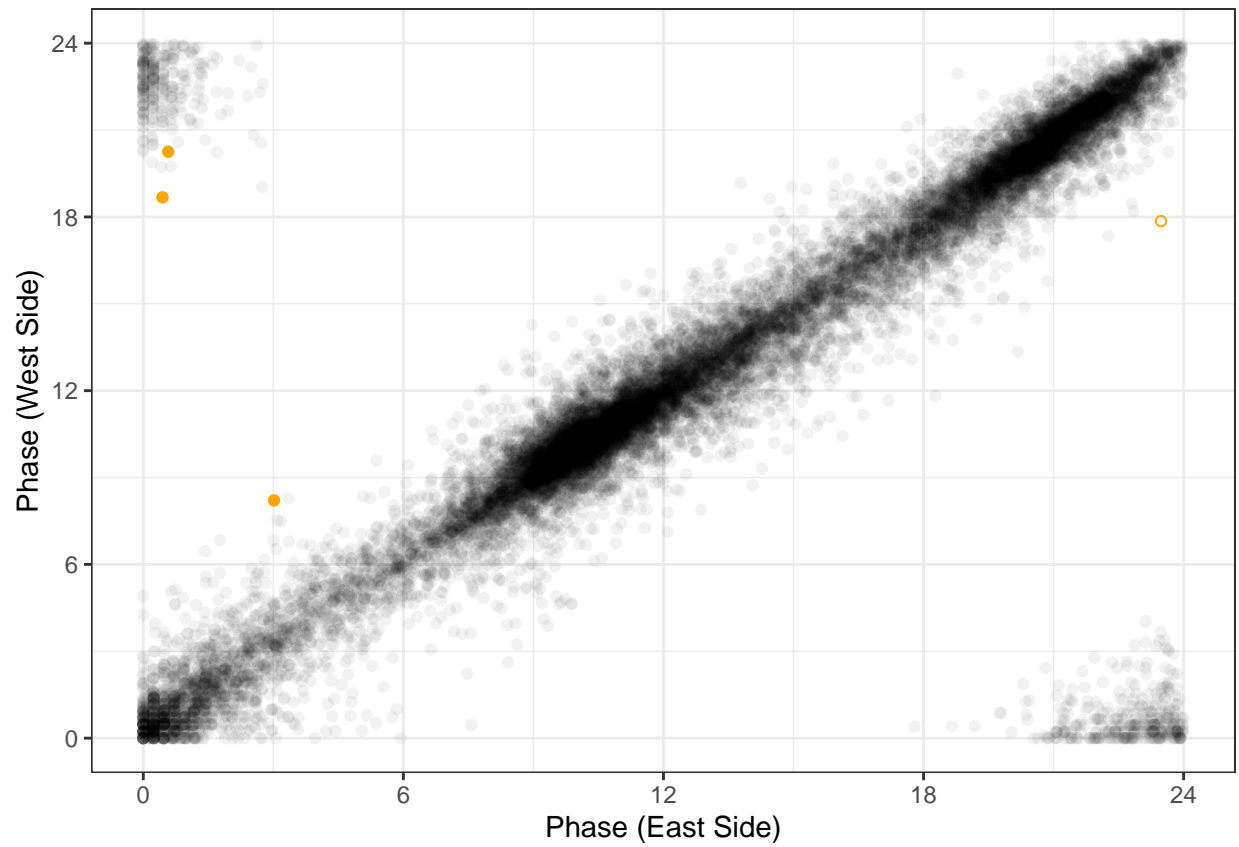
```
phase.shifted.color <- 'orange'
```

```
cosopt.both.phaseshifted <- subset(cosopt.both, GeneID %in% phase.shifted.genes)
ggplot(cosopt.both) +
  geom_point(aes(x = PeakPhase.E, y = PeakPhase.W), alpha=0.05) +
  geom_point(data = subset(cosopt, GeneID %in% phase.shifted.genes), aes(x = PeakPhase.E, y = PeakPhase
scale_x_continuous(breaks=seq(0, 24, 6)) +
scale_y_continuous(breaks=seq(0, 24, 6)) +
xlab('Phase (East Side)') +
ylab('Phase (West Side)') +
theme_bw()
```



```
ggsave('plots/phases.west-vs-east.highlight-shifted.png', w=6, h=6)
ggsave('plots/phases.west-vs-east.highlight-shifted.pdf', w=6, h=6)

cosopt.both.phaseshifted <- subset(cosopt.both, GeneID %in% phase.shifted.genes)
ggplot(cosopt.both) +
  geom_point(aes(x = PeakPhase.E, y = PeakPhase.W), alpha=0.05) +
  geom_point(data = subset(cosopt, GeneID %in% phase.shifted.genes), aes(x = PeakPhase.E, y = PeakPhase.W)) +
  geom_point(data = subset(cosopt, GeneID == 'HanXRQChr13g0402621'), aes(x = PeakPhase.E, y = PeakPhase.W)) +
  scale_x_continuous(breaks=seq(0, 24, 6)) +
  scale_y_continuous(breaks=seq(0, 24, 6)) +
  xlab('Phase (East Side)') +
  ylab('Phase (West Side)') +
  theme_bw()
```



```
ggsave('plots/phases.west-vs-east.highlight-shifted.with-SAUR14.png', w=6, h=6)
ggsave('plots/phases.west-vs-east.highlight-shifted.with-SAUR14.pdf', w=6, h=6)
```