

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 <- 0.05
min.expressed.count <- 8
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('counts/merged-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')]

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

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

# 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')

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[-1, ], 'r-data/east.counts.rds')
saveRDS(west.counts[-1, ], 'r-data/west.counts.rds')

# Get Merged Counts
west.counts.temp <- west.counts
east.counts.temp <- east.counts

colnames(west.counts.temp) <- sub('w', 'm', colnames(west.counts.temp))
colnames(east.counts.temp) <- sub('ea', 'm', colnames(east.counts.temp))

gene.ids <- rownames(counts)
merged.sample.ids <- intersect(colnames(west.counts.temp), colnames(east.counts.temp))
merged.counts = data.frame(matrix(vector(), length(gene.ids),
  length(merged.sample.ids), dimnames=list(gene.ids, merged.sample.ids)),
  stringsAsFactors=F)

for (sample.id in merged.sample.ids) {
  merged.counts[, colnames(merged.counts) == sample.id] <- rowMeans(cbind(
    west.counts.temp[, colnames(west.counts.temp) == sample.id],
    east.counts.temp[, colnames(east.counts.temp) == sample.id]
  ))
}

```

```

write.table(merged.counts, 'counts/merged-counts.tsv', sep='\t', quote=F, col.names=F)
saveRDS(merged.counts[-1, ], 'r-data/merged.counts.rds')

# Prepare timecourse for plotting
timecourse <- data.frame(hour, side, t(counts))
timecourse.sides <- data.frame(hour, side, t(counts[rownames(counts) != "Gene", ]))
hour.merged <- as.numeric(merged.counts[rownames(merged.counts) == "Gene", ])
timecourse.merged <- data.frame(hour = hour.merged, side = "Merged", t(merged.counts[rownames(merged.counts) == "Gene", ]))
timecourse.all <- rbind(timecourse.sides, timecourse.merged)
timecourse.all <- melt(timecourse.all, id.vars=c('hour', 'side'), variable.name='gene', value.name='counts')
timecourse <- ddpby(timecourse.all, .(hour, side, gene), summarize, mean=mean(counts), stderr=sqrt(var(counts)))
saveRDS(timecourse, 'r-data/timecourse.rds')
}

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

timecourse.summary.mean <- dcast(timecourse, gene ~ side + hour, value.var = "mean")
timecourse.summary.stderr <- dcast(timecourse, gene ~ side + hour, value.var = "stderr")
timecourse.summary <- merge(timecourse.summary.mean, timecourse.summary.stderr, by = 'gene', all = TRUE)
names(timecourse.summary)[names(timecourse.summary) == 'gene'] <- 'GeneID'

```

## 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',
                             merged.color='black', plot.merged=FALSE,
                             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, ncol=1, .timecourse=timecourse) {

  library(ggplot2)
  timecourse.subset <- .timecourse[.timecourse$gene %in% gene.list, ]
  if (plot.merged) {
    plot.colors <- c(east.color, west.color, merged.color)
  } else {
    plot.colors <- c(east.color, west.color)
    timecourse.subset <- subset(timecourse.subset, side != "Merged")
  }
  timecourse.subset$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)

  if (plot.merged) {
    p <- p + geom_line(data=subset(timecourse.subset, side=='Merged'), aes(x=hour, y=mean), color='white', size=1)
  }

  p <- p +
    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)

  if (plot.merged) {
    p <- p + geom_errorbar(data=subset(timecourse.subset, side=='Merged'), 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=plot.colors)

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=ncol, scales='free_y')
}

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

```

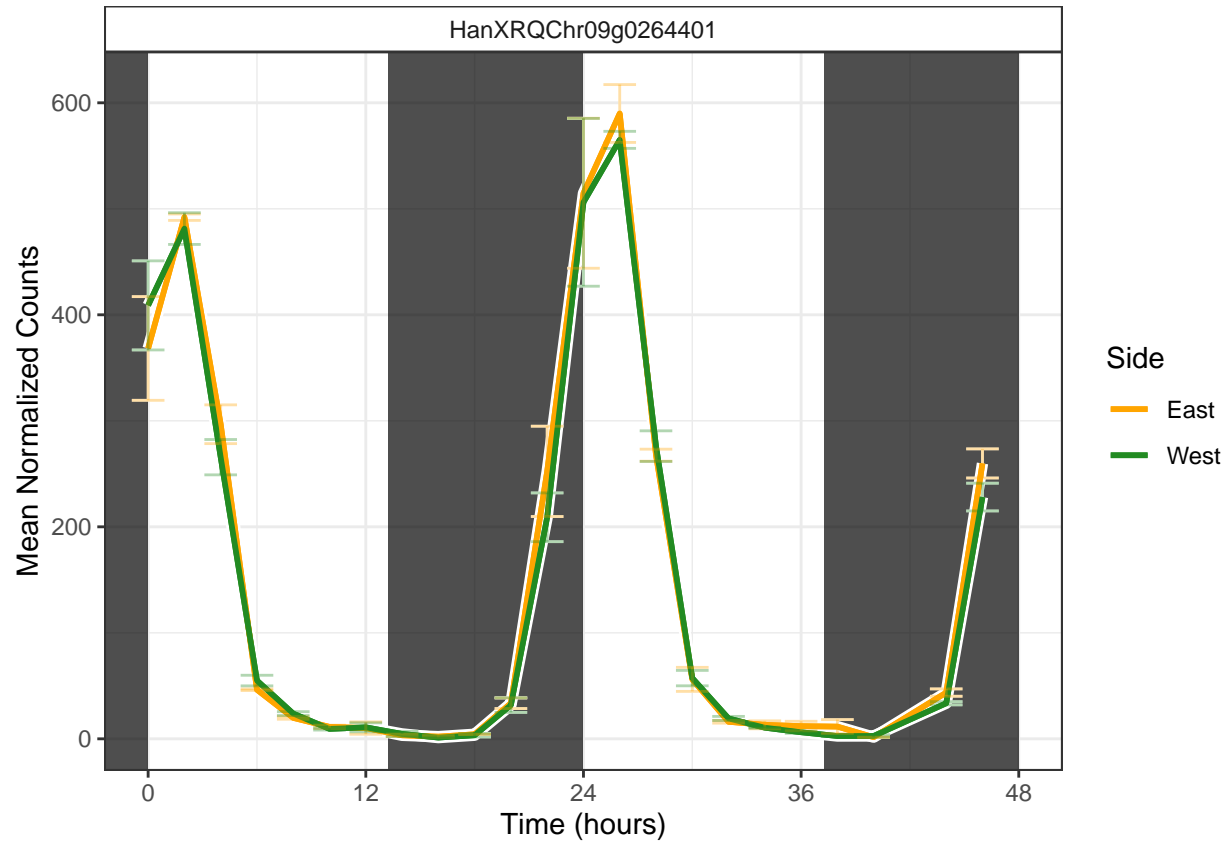
```

}

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

demo.gene.list <- c('HanXRQChr09g0264401', 'HanXRQChr15g0489581', 'HanXRQChr04g0118841', 'HanXRQChr01g0
# Plot single gene
plot.timecourse(demo.gene.list[1], lights.off=13.25)

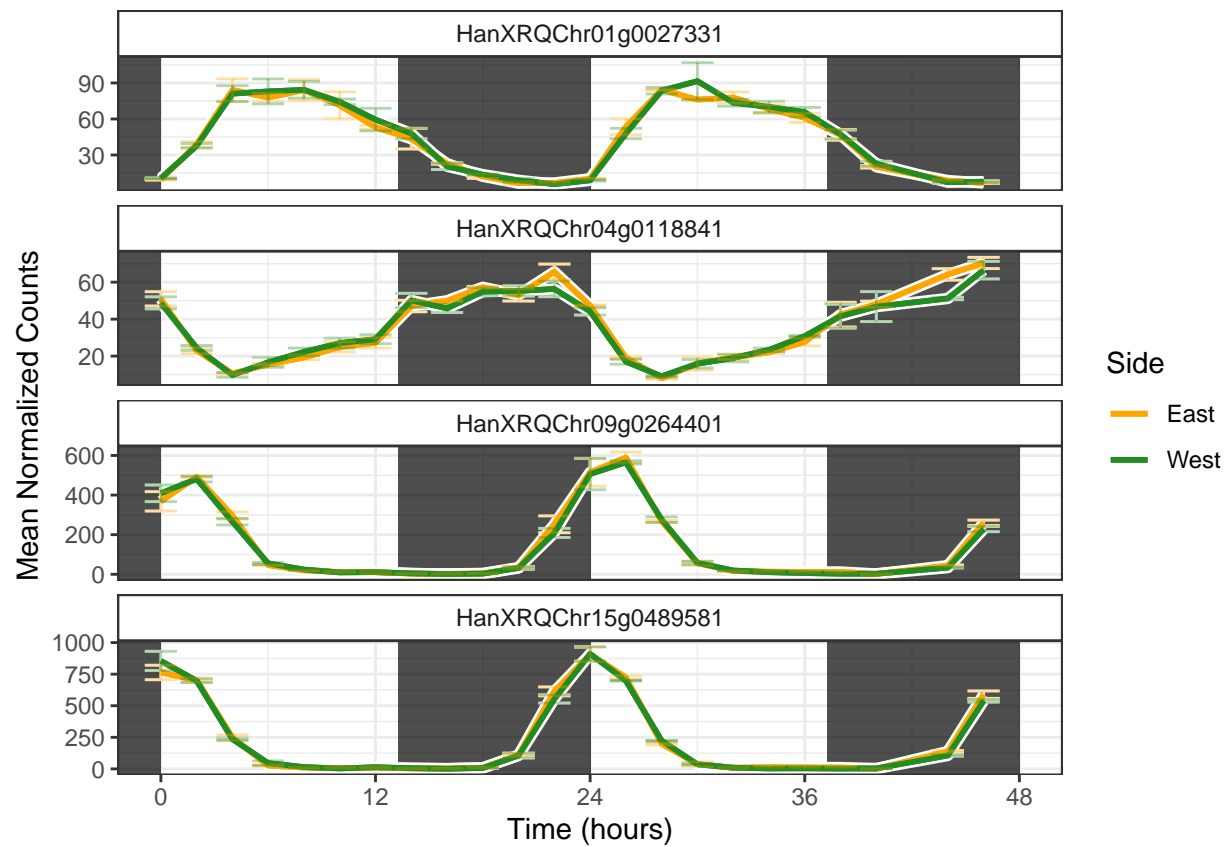
```



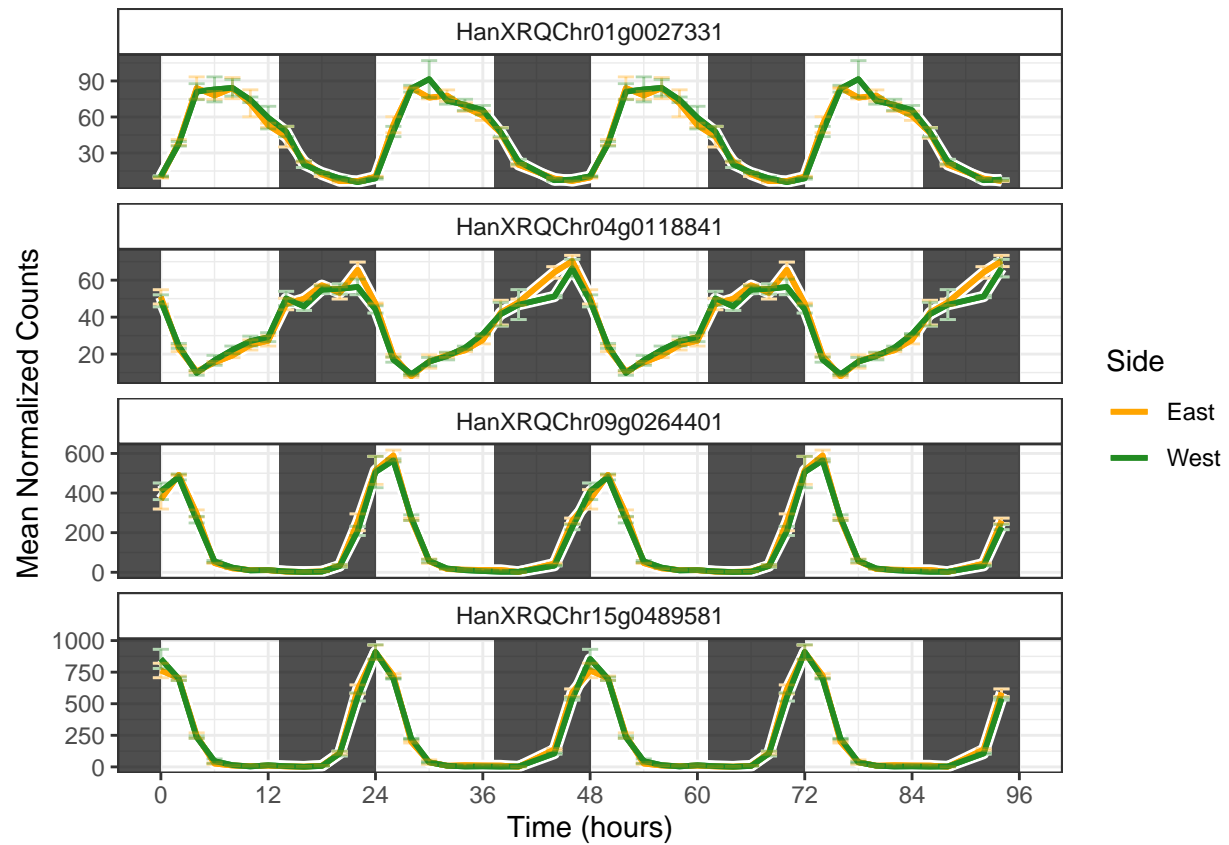
```

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

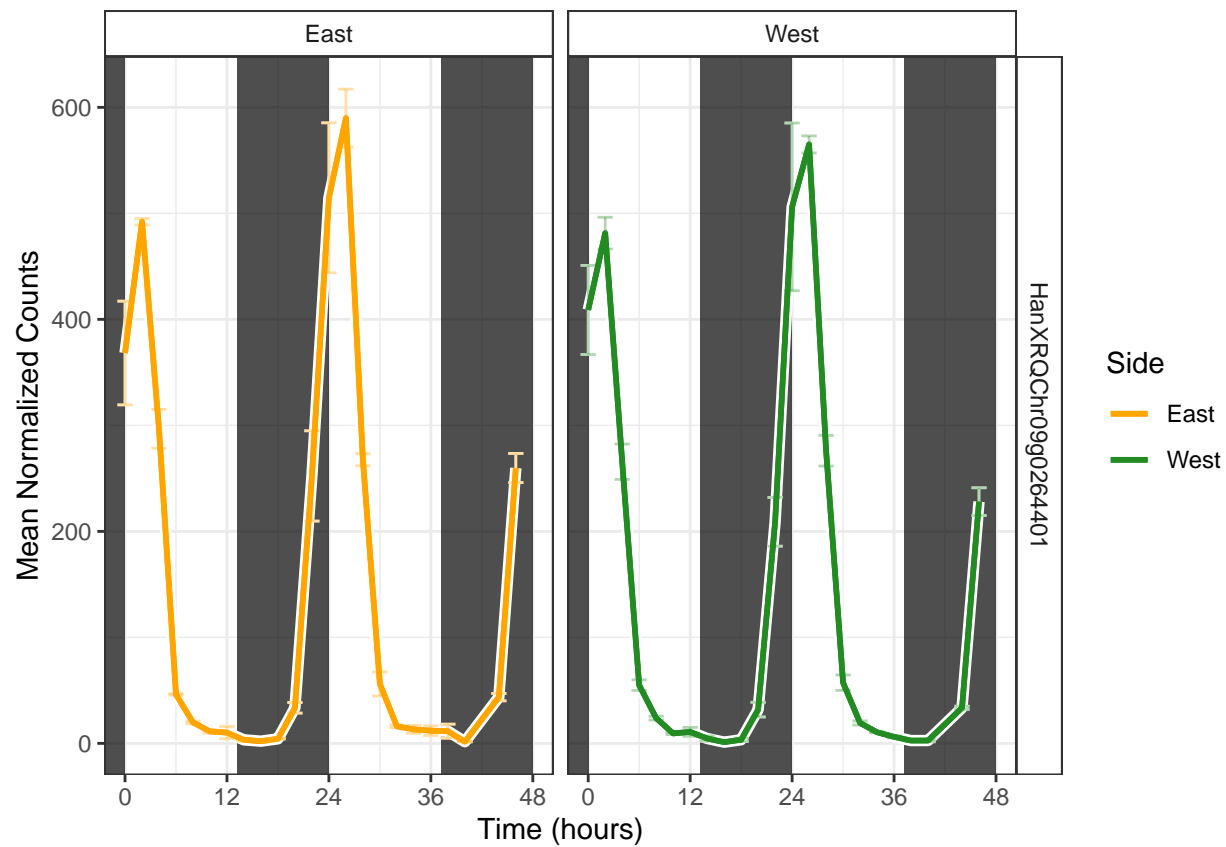
```



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

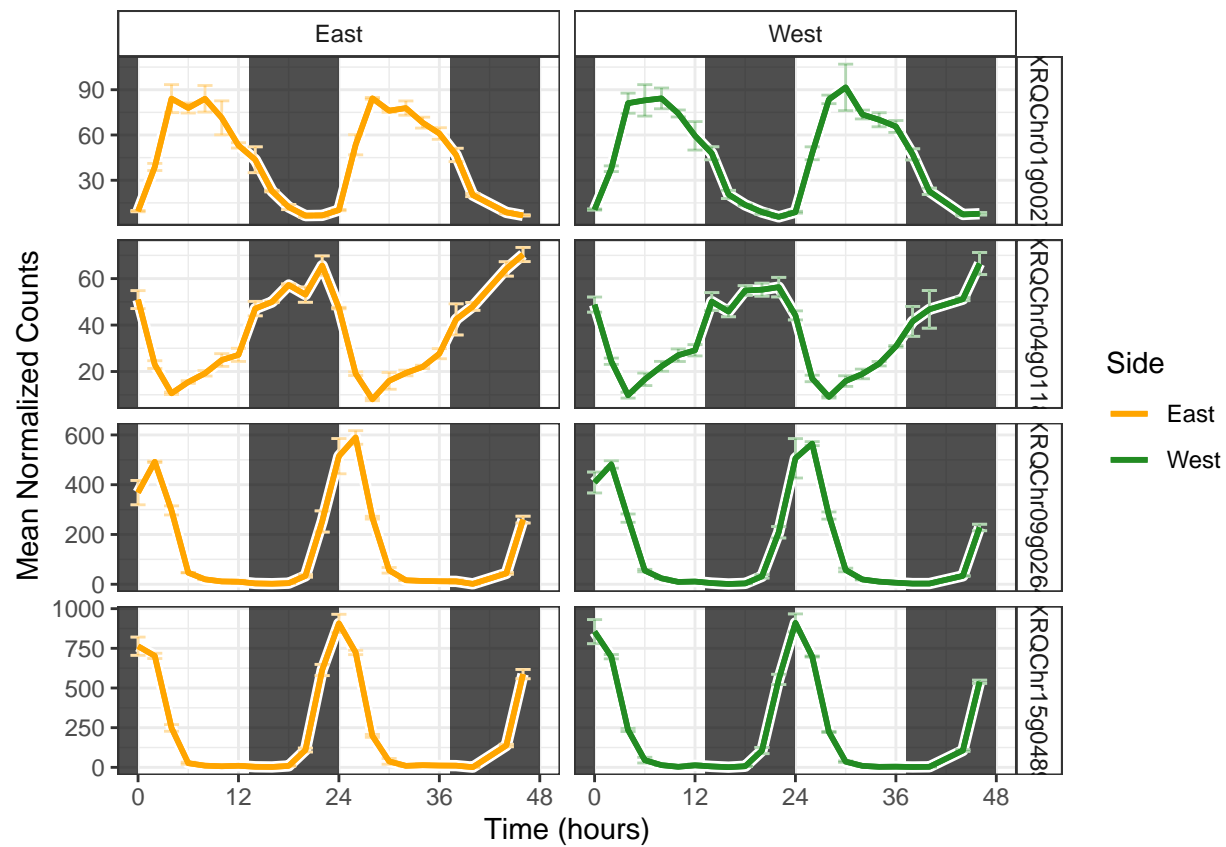


```
# Plot side-by-side
plot.timecourse(demo.gene.list[1], lights.off=13.25, side.by.side=TRUE)
```

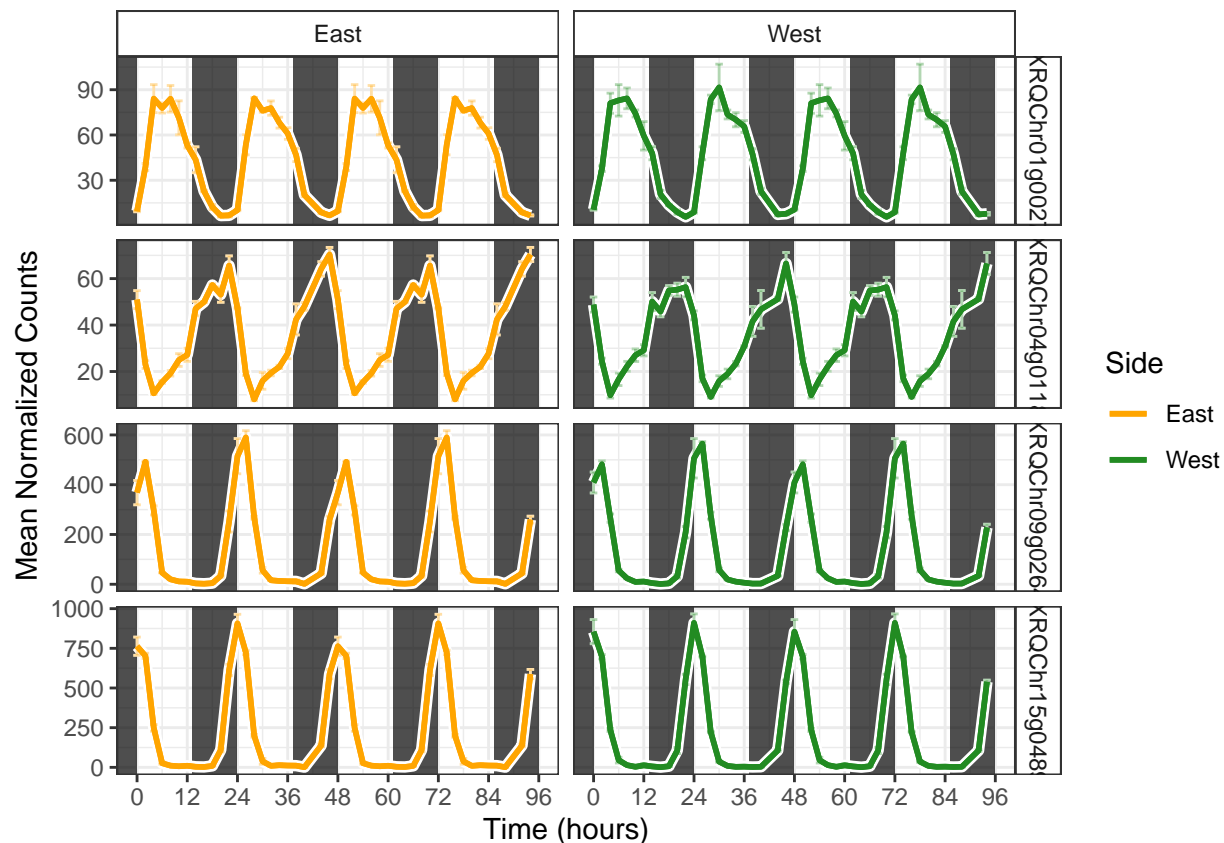


```
plot.timecourse(demo.gene.list, lights.off=13.25, side.by.side=TRUE)
```





```
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

get.expressed.genes <- function(min.meanexplev = NULL, min.expressed.count = NULL) {
  if(is.null(min.meanexplev)) stop("No minimum expression level given.")

  mean.expression <- data.frame(
    east = rowMeans(timecourse.summary.mean[, grepl("East", names(timecourse.summary.mean))]),
    west = rowMeans(timecourse.summary.mean[, grepl("West", names(timecourse.summary.mean))]),
    merged = rowMeans(timecourse.summary.mean[, grepl("Merged", names(timecourse.summary.mean))])
  )
  rownames(mean.expression) <- timecourse.summary.mean$gene

  if(is.null(min.expressed.count)) {
    expressed.genes <- as.data.frame(mean.expression >= min.meanexplev)
    rownames(expressed.genes) <- rownames(mean.expression)
  } else {
    expressed.frequency <- data.frame(
      east = rowSums(timecourse.summary.mean[, grepl("East", names(timecourse.summary.mean))] > min.meanexplev)
      west = rowSums(timecourse.summary.mean[, grepl("West", names(timecourse.summary.mean))] > min.meanexplev)
      merged = rowSums(timecourse.summary.mean[, grepl("Merged", names(timecourse.summary.mean))] > min.meanexplev)
    )
  }
}

```

```

rownames(expressed.frequency) <- timecourse.summary.mean$gene
expressed.genes <- as.data.frame(expressed.frequency >= min.expressed.count)
rownames(expressed.genes) <- rownames(expressed.frequency)
}

out <- list()
out$mean.expression <- mean.expression
out$expressed.genes <- expressed.genes
return(out)
}

```

```

expressed.genes <- get.expressed.genes(
  min.meanexplev = min.meanexplev, min.expressed.count = min.expressed.count)
expressed <- expressed.genes$expressed
mean.expression <- expressed.genes$mean.expression

```

```

#Expressed in East: 40,291
sum(expressed$east)

```

```
## [1] 40291
```

```

#Expressed in West: 40,354
sum(expressed$west)

```

```
## [1] 40354
```

```

#Expressed in Merged: 40,228
sum(expressed$merged)

```

```
## [1] 40228
```

```

#Expressed in East or West: 40,739
sum(expressed$east | expressed$west)

```

```
## [1] 40739
```

```

#Expressed in East and West: 39,906
sum(expressed$east & expressed$west)

```

```
## [1] 39906
```

```

#Expressed in East, West, and Merged: 39,832
sum(expressed$east & expressed$west & expressed$merged)

```

```
## [1] 39832
```

```

#Expressed in East, West, or Merged: 40,781
sum(expressed$east | expressed$west | expressed$merged)

```

```
## [1] 40781
```

```

# Get rhythmic genes

```

```

rhythmic.east <- as.character(cosopt.east$GeneID[cosopt.east$pMMC.Beta < min.p.mmc.beta & cosopt.east$G
rhythmic.west <- as.character(cosopt.west$GeneID[cosopt.west$pMMC.Beta < min.p.mmc.beta & cosopt.west$G
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)
rhythmic.any <- union(rhythmic.merged, union(rhythmic.east, rhythmic.west))

```

```

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

## [1] 22005
# [1] 22005
length(intersect(rhythmic.merged, rhythmic.west))

## [1] 22014
# [1] 22014

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

length(rhythmic.east)

## [1] 23083
# [1] 23083
length(rhythmic.west)

## [1] 23172
# [1] 23172
length(rhythmic.merged)

## [1] 25778
# [1] 25778

length(rhythmic.both)

## [1] 19447
# [1] 19447
length(rhythmic.all)

## [1] 19409
# [1] 19409
length(rhythmic.any)

## [1] 27976
# [1] 27976

length(rhythmic.east.only)

## [1] 3636
# [1] 3636
length(rhythmic.west.only)

## [1] 3725
# [1] 3725

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 Expressed Genes:

```
East: 40,291
West: 40,354
East or West: 40,739
East and West: 39,906
Merged: 40,228
East, West, or Merged: 39,832
East, West, and Merged: 40,781
```

Rhythmic Genes in East or West time course: 26,808

```
East only: 3,636 (13.6%)
West only: 3,725 (13.9%)
Both East and West: 19,447 (72.5%)
```

Rhythmic Genes in Merged time course: 25,778

Rhythmic Genes in any time course (East, West, and Merged): 27,976

Rhythmic Genes in all three time courses (East, West, and Merged): 19,409

	Rhythmic	Expressed	% Rhythmic
East	23,083	40,291	57.3%
West	23,172	40,354	57.4%
East or West	26,808	40,739	65.8%
East and West	19,447	39,906	48.7%
Merged	25,778	40,228	64.1%
East, West, or Merged	27,976	39,832	70.2%
East, West, and Merged	19,409	40,781	47.6%

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()

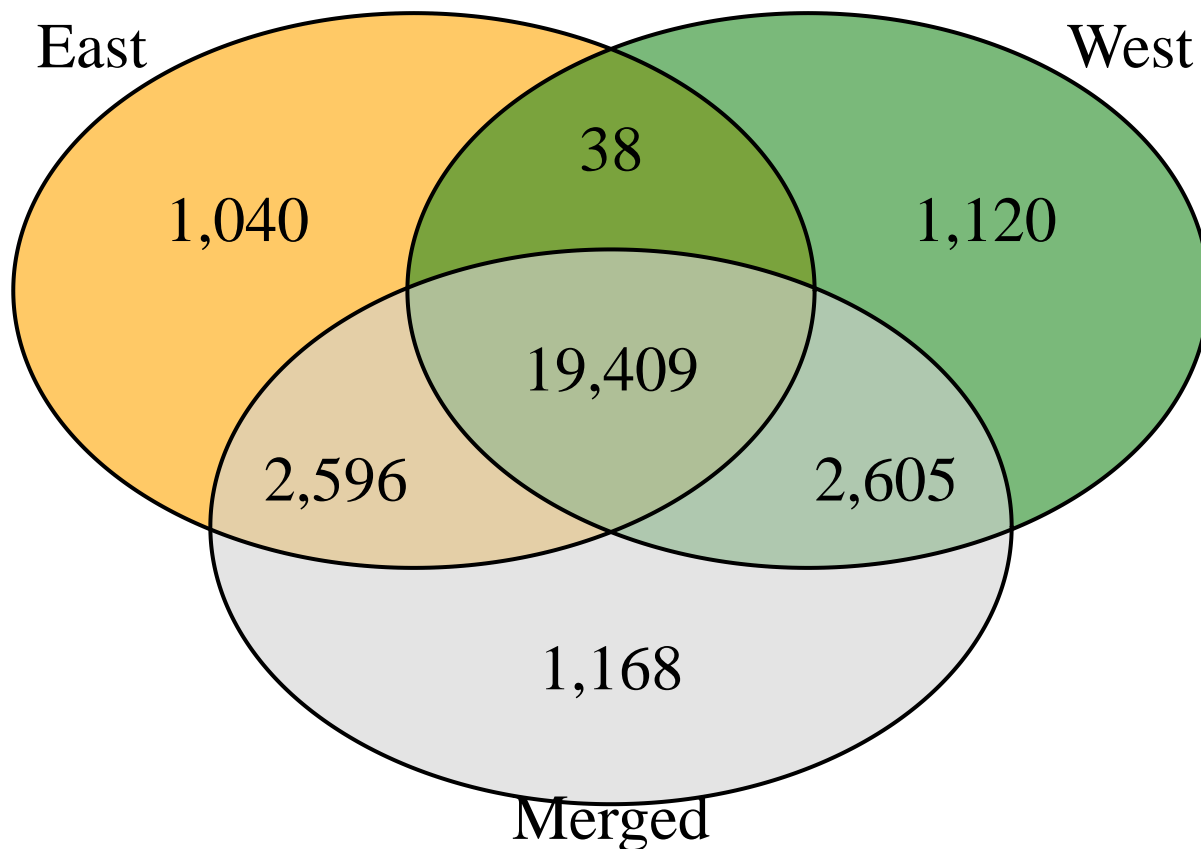
## pdf
## 2

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

## pdf
## 2

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

```



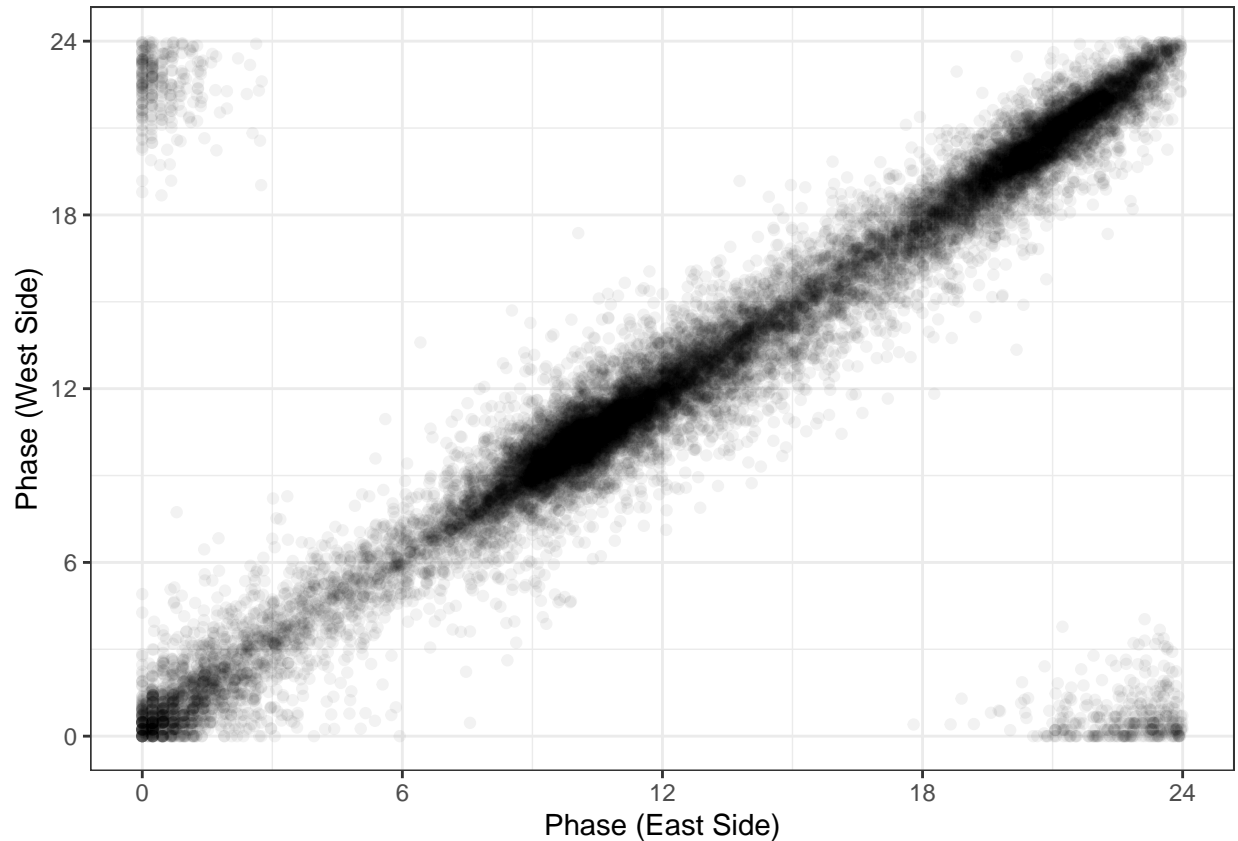
West vs East Phase

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

```
## [1] -0.0003668416
```

```
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, useDingbats = FALSE)
```

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)/length(TempC)))
```

```
## |
temperature.stats.scaled <- ddply(temperatures, .(Time), summarize, mean=mean(ScaledTempC), stderr=sqrt
```

```
## |
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
```

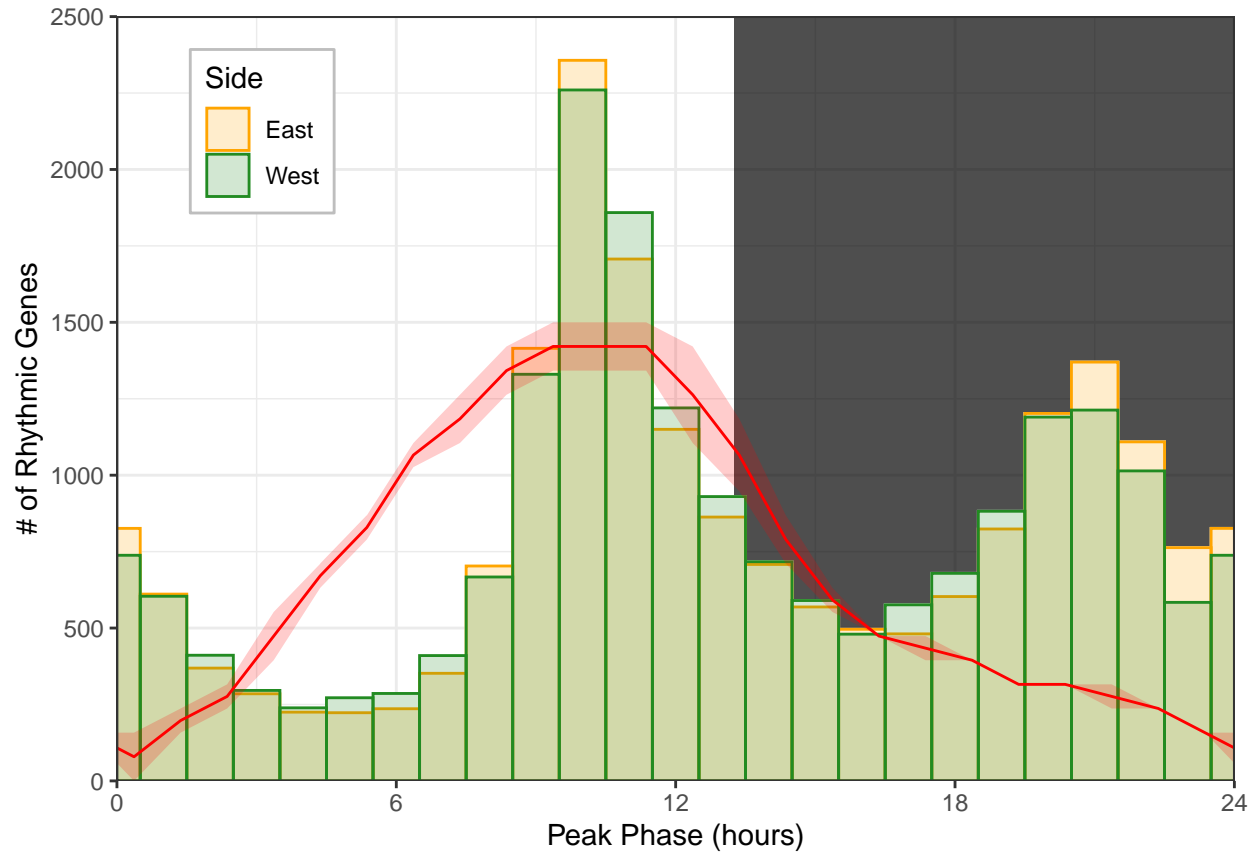
```
temperature.stats
```

```
##      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

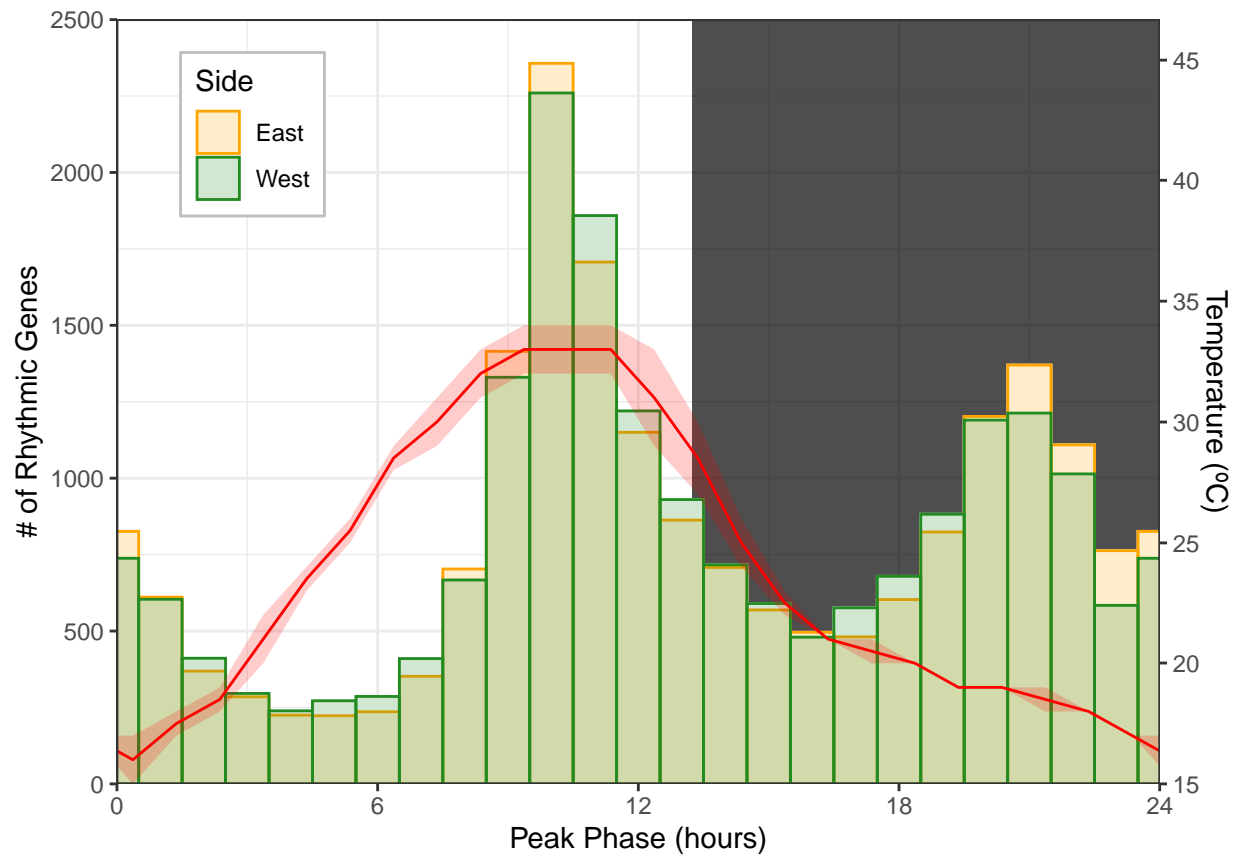
```
p <- ggplot() +
  geom_rect(data=daynight, aes(xmin=dawn, xmax=dusk), fill='black', ymin=-10000, ymax=10000, alpha=0.7) +
  geom_histogram(data=subset(histogram.data.combined, side=='West'), aes(x=PeakPhase, y=..count..), color='red', fill='red', alpha=0.5) +
  geom_histogram(data=subset(histogram.data.combined, side=='East'), aes(x=PeakPhase, y=..count..), color='green', fill='green', alpha=0.5) +
  geom_histogram(data=histogram.data.combined, aes(x=PeakPhase, color=side, fill=side, y=..count..), alpha=0.5) +
  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 = '# of Rhythmic Genes') +
  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() +
  theme(legend.position = c(.13, .85), legend.background = element_rect(linetype = 'solid', colour = 'green'))
```

p



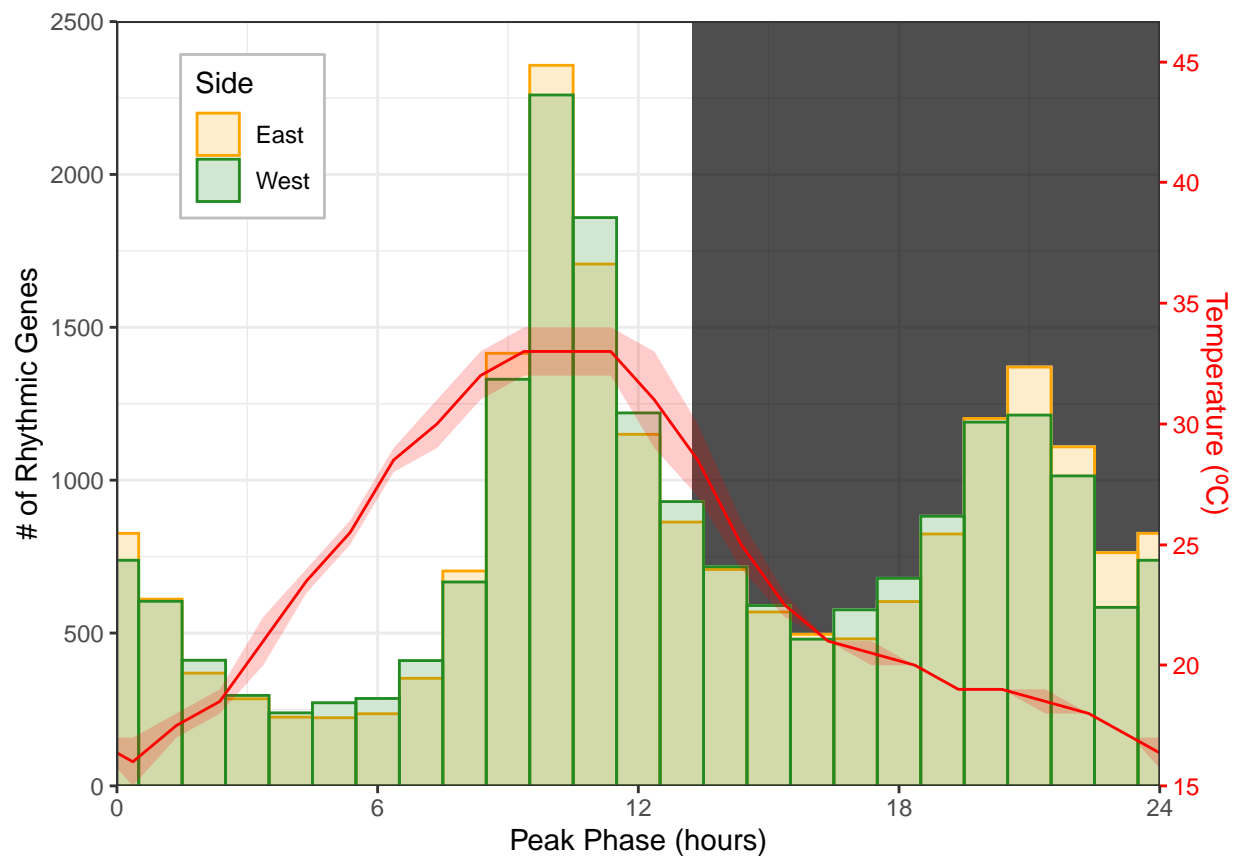
```
ggsave('plots/phase-histogram.temperature.png', w=6, h=5)
ggsave('plots/phase-histogram.temperature.pdf', w=6, h=5, useDingbats = FALSE)

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(
```



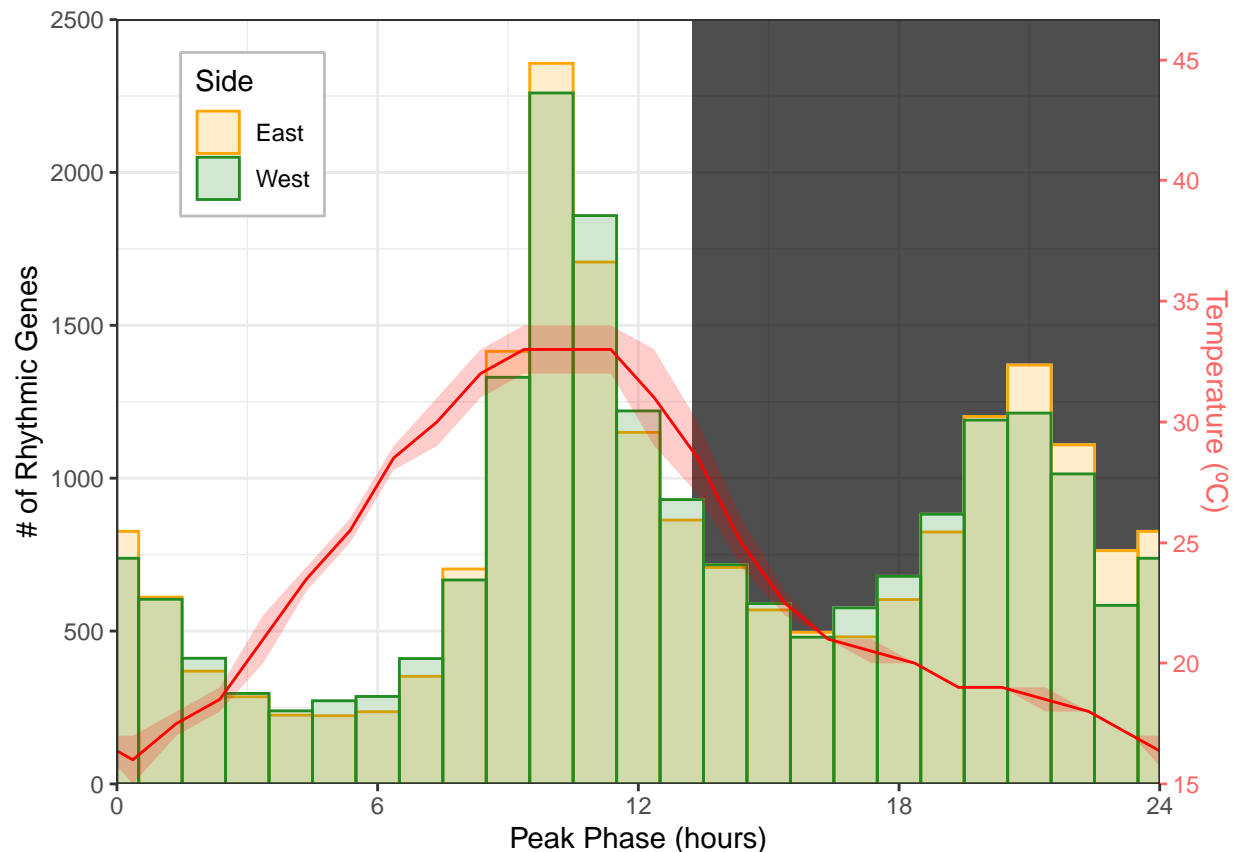
```
ggsave('plots/phase-histogram.temperature-axis.png', w=6, h=5)
ggsave('plots/phase-histogram.temperature-axis.pdf', w=6, h=5, useDingbats = FALSE)

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, useDingbats = FALSE)

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, useDingbats = FALSE)
```

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
## "713a7ed174948290a41708e29e9d65ab"
```

## Plot Amplitude Differences Summary

```
plot.ampdiff.summary <- function() {
  timecourse.subset <- subset(timecourse, side != "Merged")
  timecourse.w <- subset(timecourse.subset, gene %in% west.high)
  timecourse.e <- subset(timecourse.subset, 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)
```

```

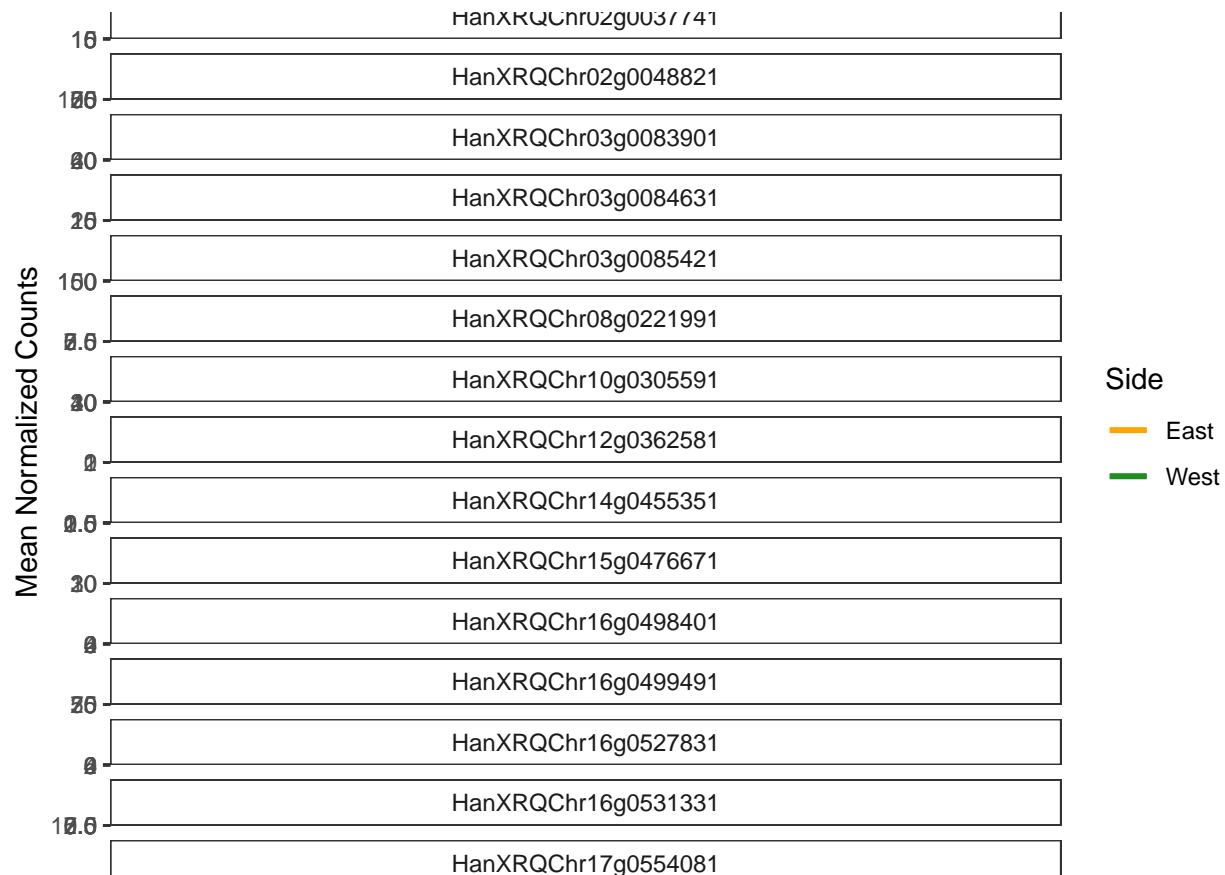
timecourse.e <- melt(timecourse.e, id.vars='hour', variable.name='side', value.name='mean.norm', na.rm=TRUE)

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 | MeanExpLev.E > 0.5))
plot.timecourse(expdiff$GeneID, lights.off = 13.25)

```



```

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

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

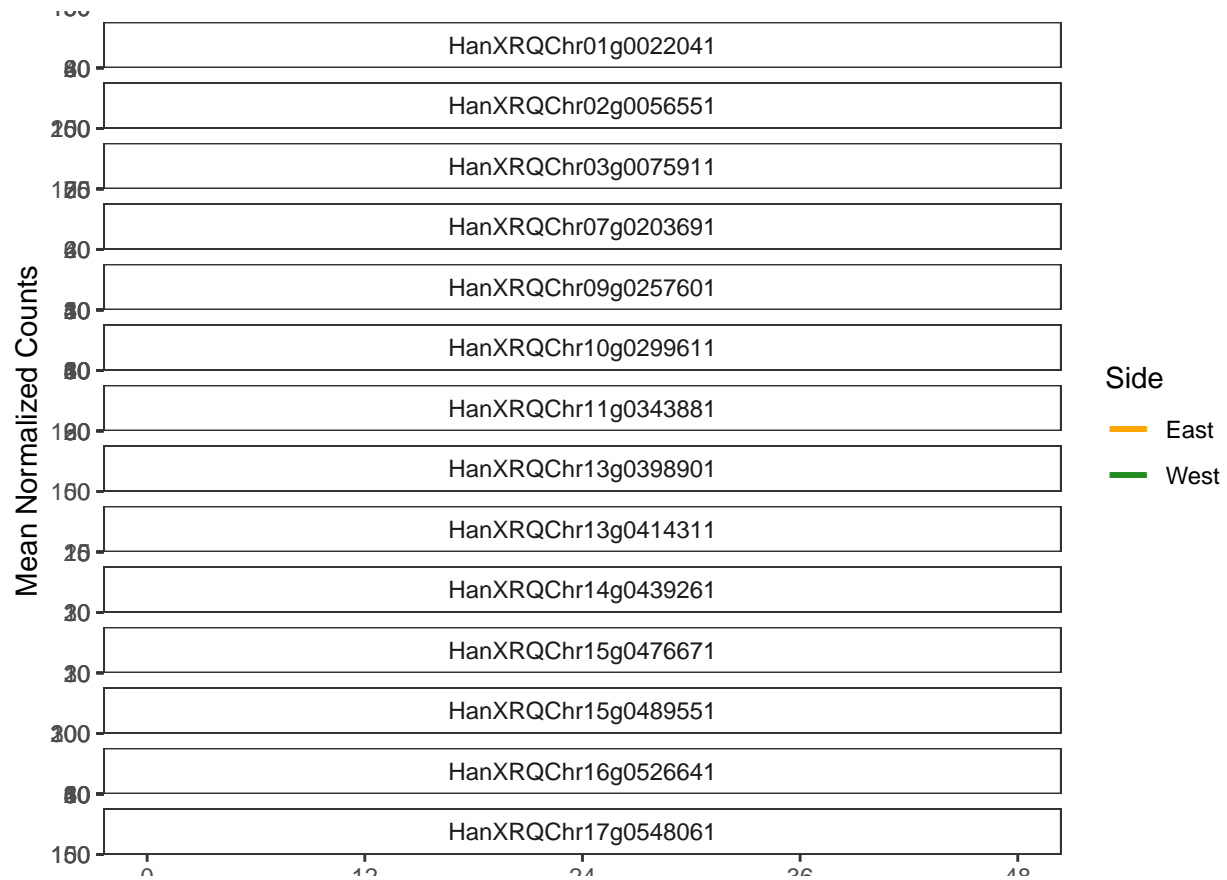
```



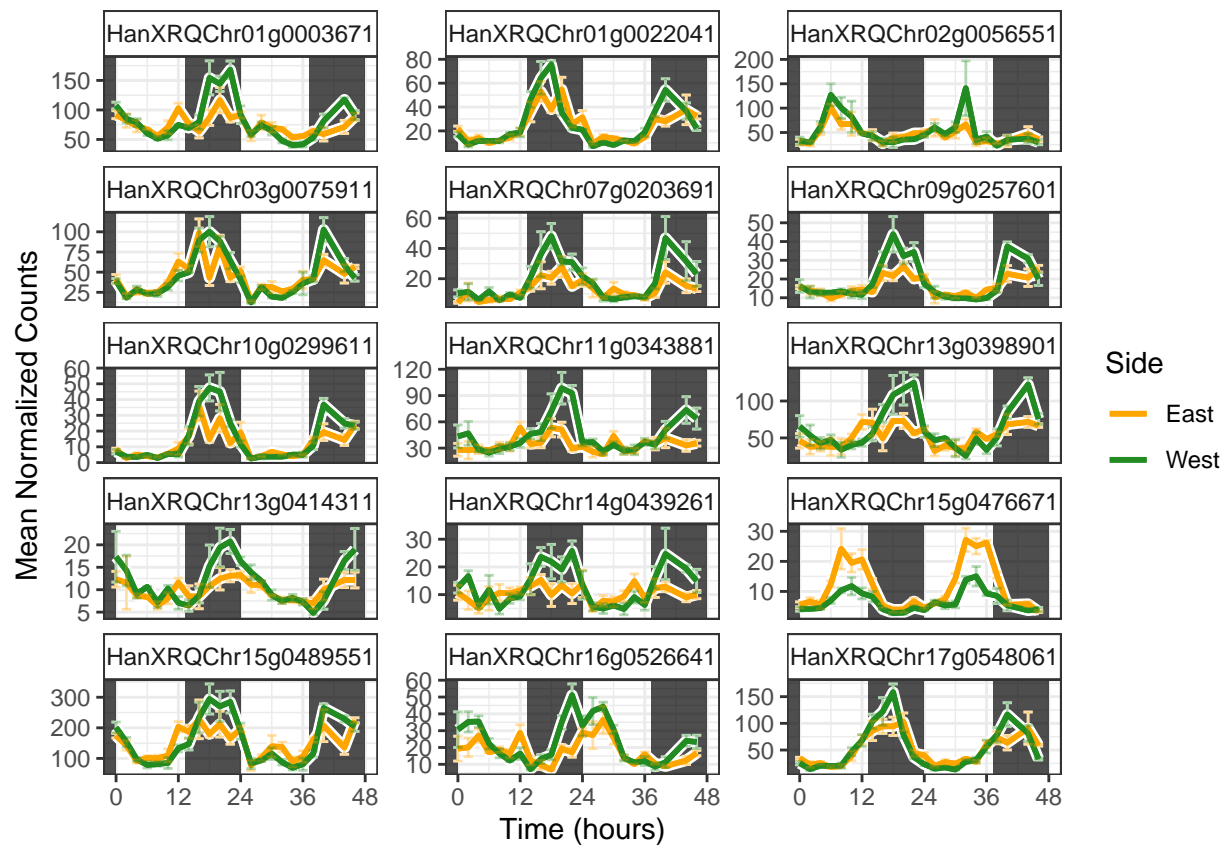
```
ampdiff <- subset(cosopt, GeneID %in% rhythmic.both & abs(amp.diff) > 0.25 & (MeanExpLevel.E > 10 | MeanExpLevel.W > 10))
amp <- rownames(ampdiff)
```

```
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('plots/amp-diff.png', w=6, h=23)
ggsave('plots/amp-diff.pdf', w=6, h=23, useDingbats = FALSE)
plot.timecourse(amp, lights.off = 13.25, ncol = 3)
```



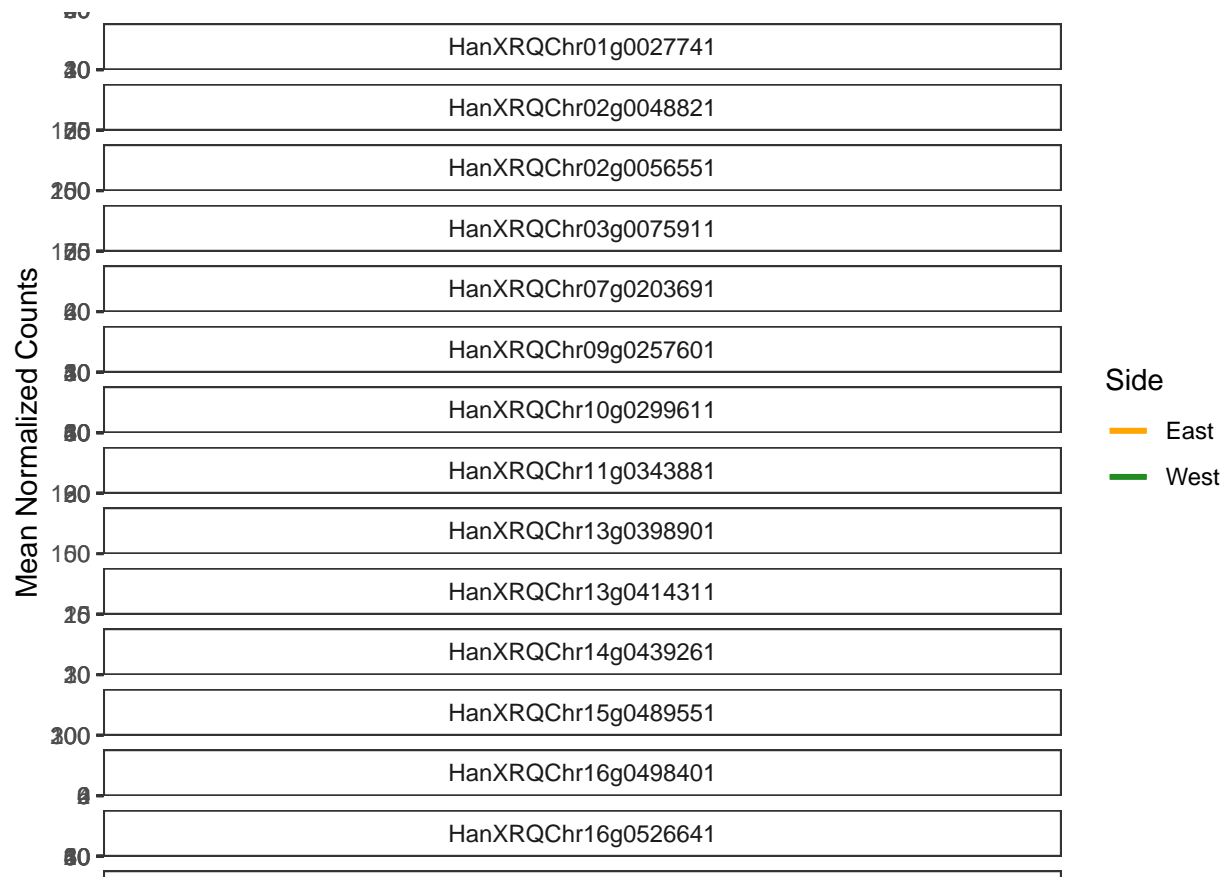
```

ggsave('plots/amp-diff.3col.standard-size.png', h=6.35, w=7.5)
ggsave('plots/amp-diff.3col.standard-size.pdf', h=6.35, w=7.5, useDingbats = FALSE)
write.table(ampdiff, 'cosopt-processed/cosopt-processed.amp-diff.txt', sep = "\t", quote = FALSE, col.names = TRUE)

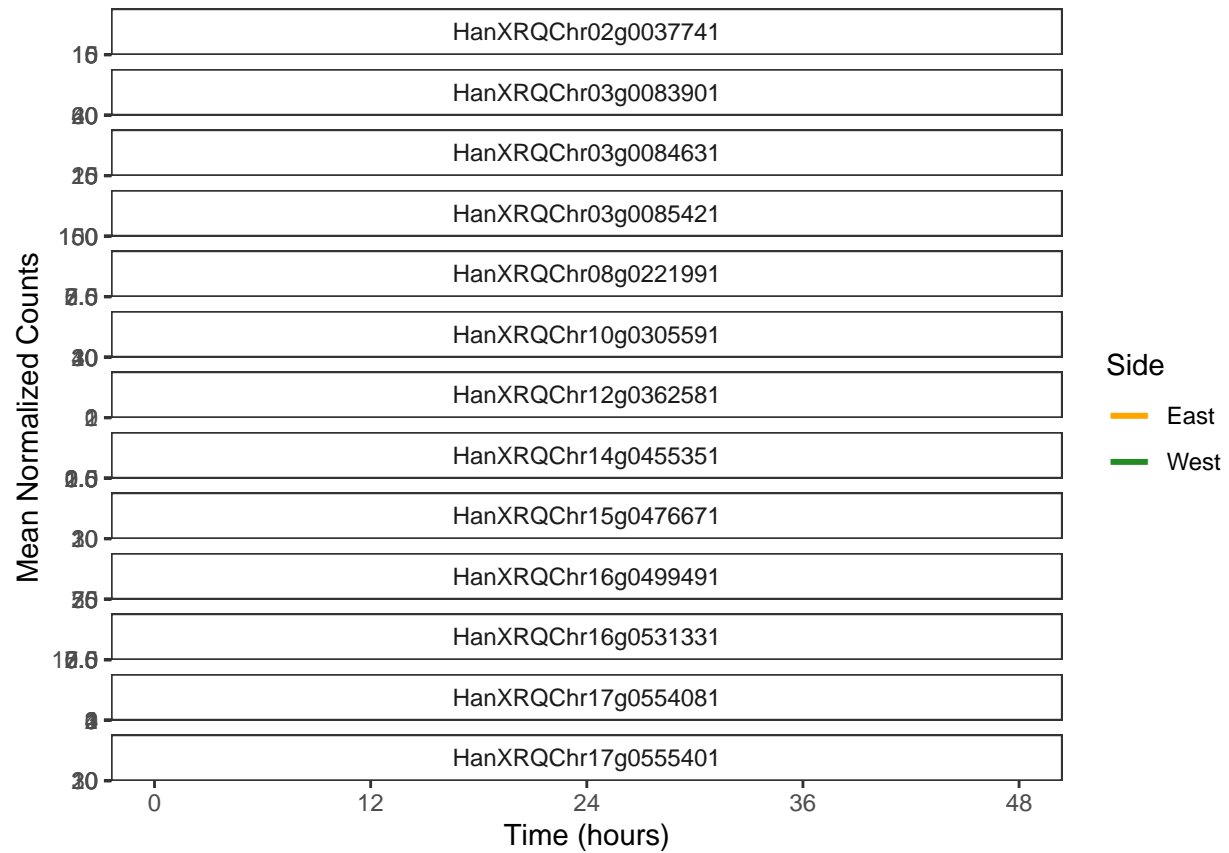
west.high <- union(exp.w, amp.w)
east.high <- union(exp.e, amp.e)

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

```

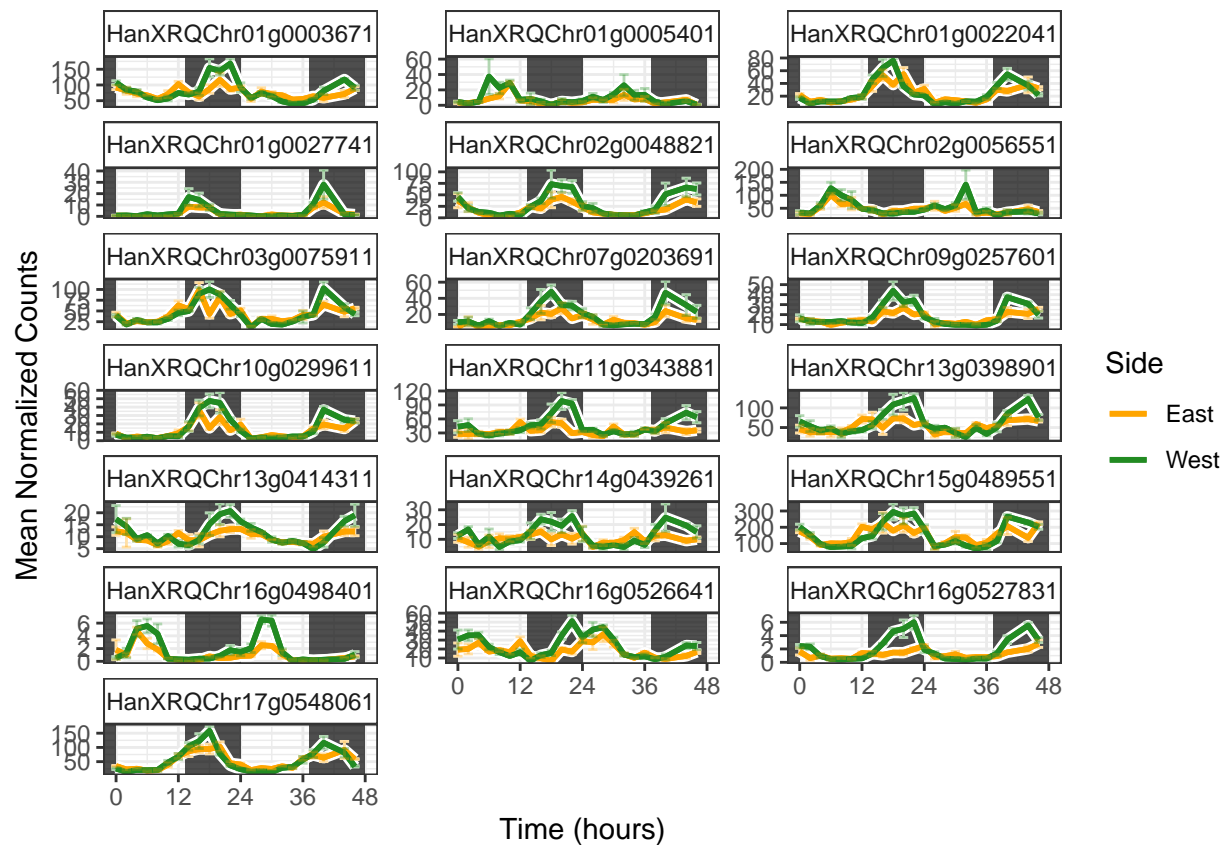


```
ggsave('plots/amp-exp-diff.west-high.png', w=6, h=30)
ggsave('plots/amp-exp-diff.west-high.pdf', w=6, h=30, useDingbats = FALSE)
plot.timecourse(east.high, lights.off = 13.25)
```

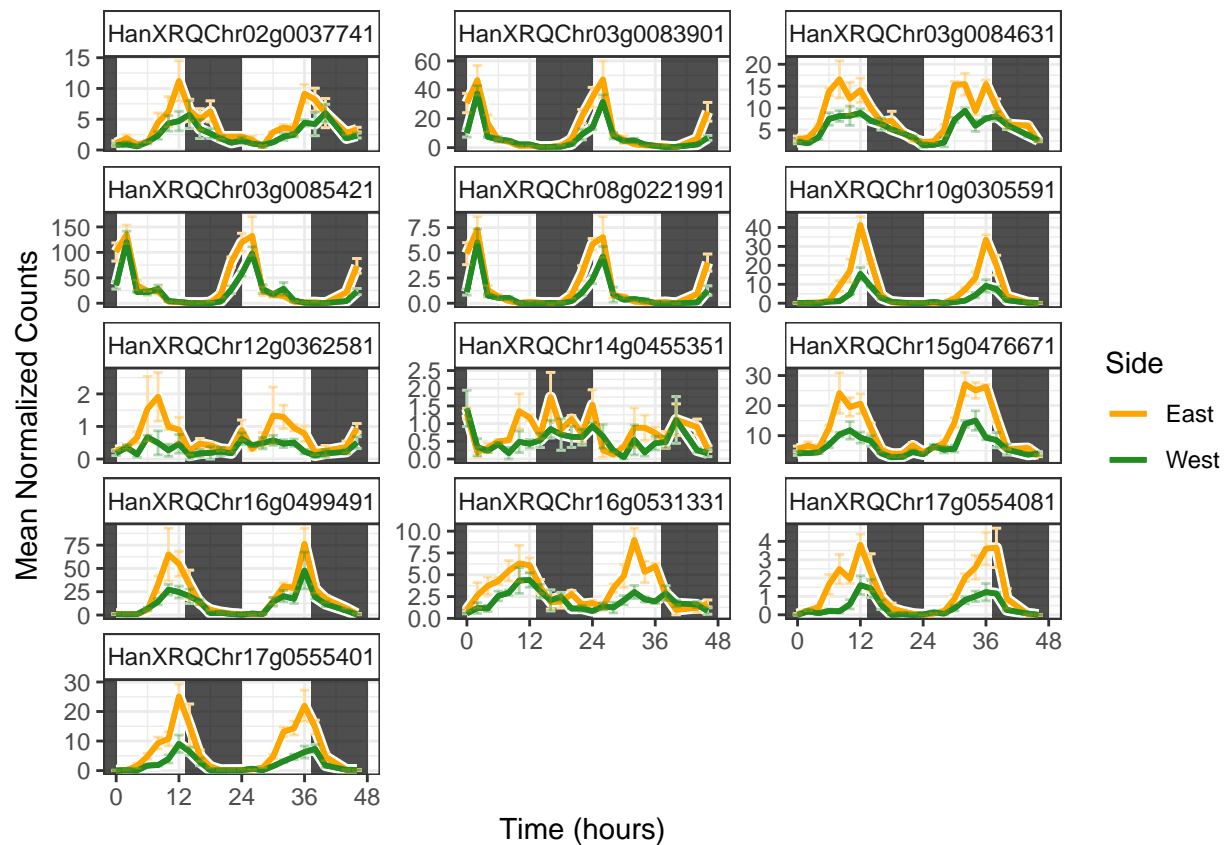


```
ggsave('plots/amp-exp-diff.east-high.png', w=6, h=21)
ggsave('plots/amp-exp-diff.east-high.pdf', w=6, h=21, useDingbats = FALSE)

plot.timecourse(west.high, lights.off = 13.25, ncol=3)
```

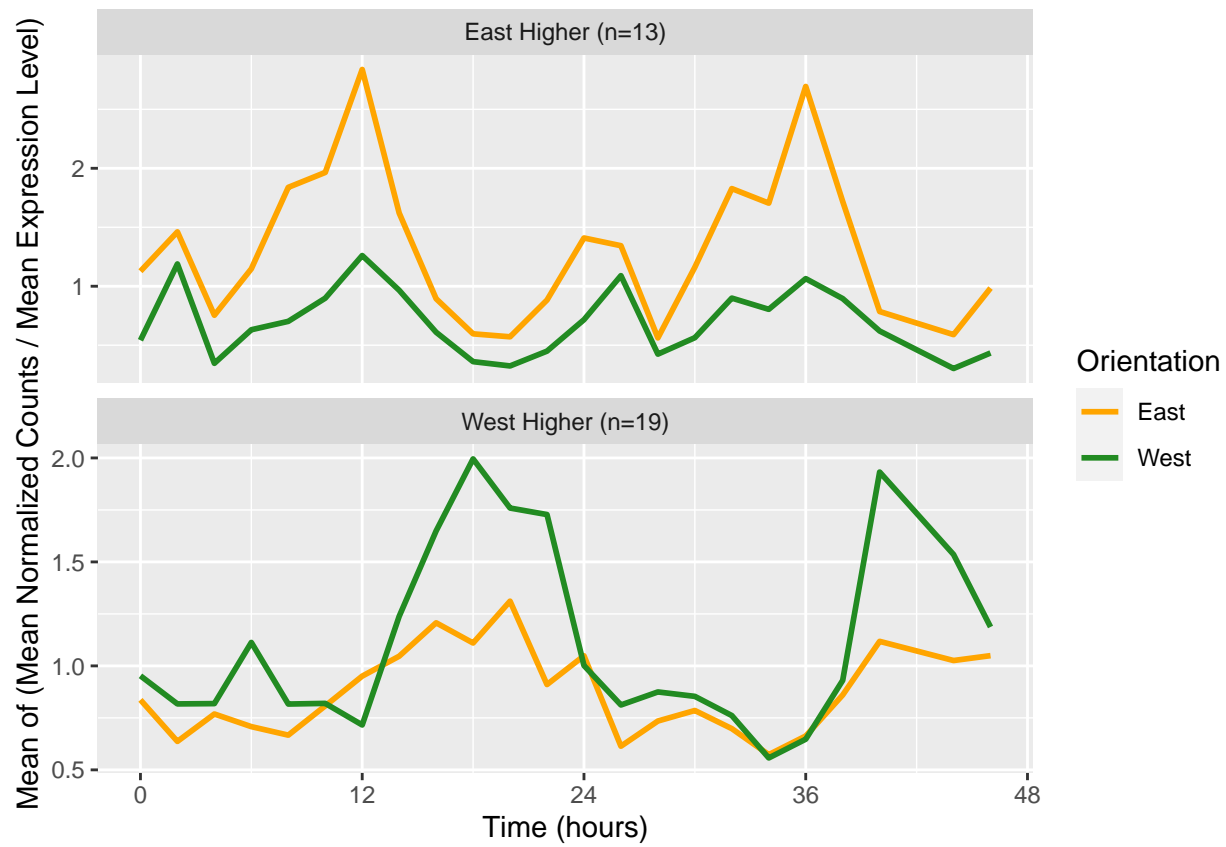


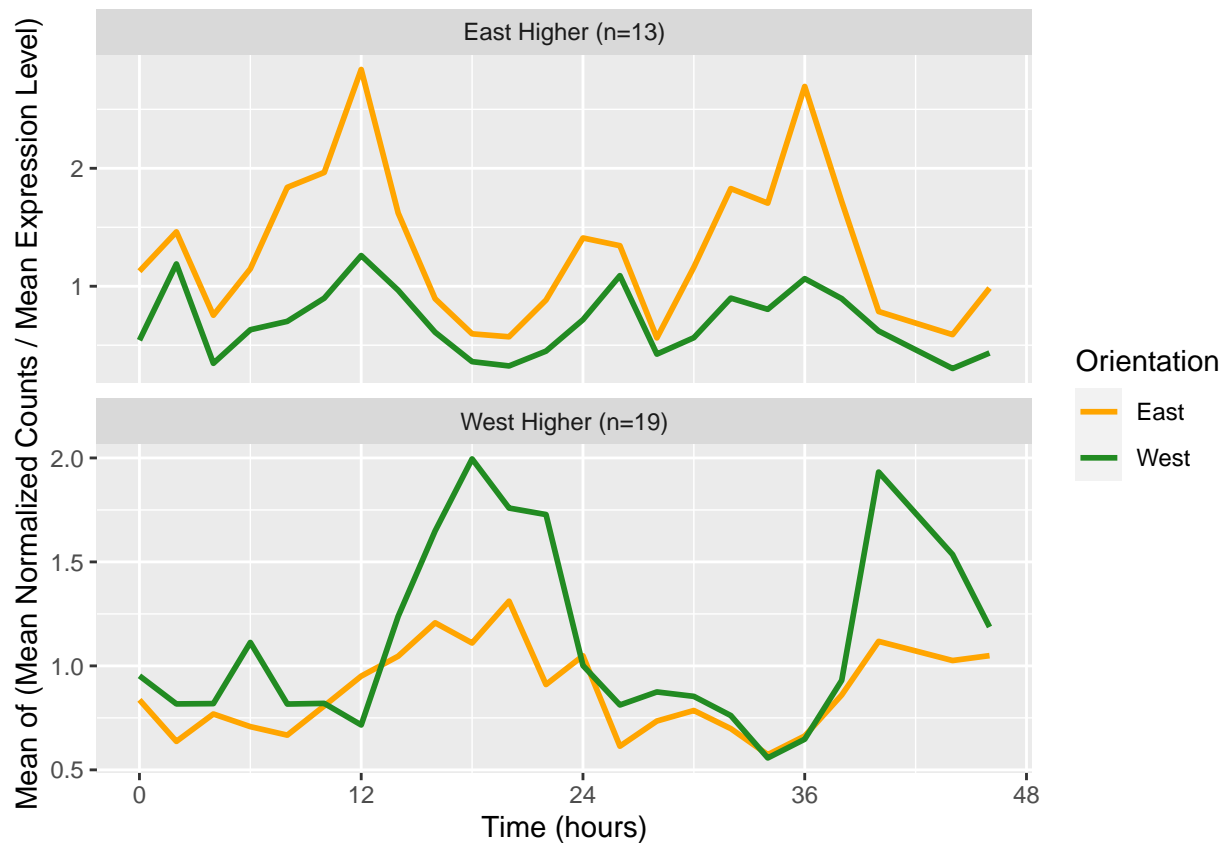
```
ggsave('plots/amp-exp-diff.west-high.3col.png', h=8.75, w=7.5)
ggsave('plots/amp-exp-diff.west-high.3col.pdf', h=8.75, w=7.5, useDingbats = FALSE)
plot.timecourse(east.high, lights.off = 13.25, ncol=3)
```



```
ggsave('plots/amp-exp-diff.east-high.3col.png', h=8.75, w=7.5)
ggsave('plots/amp-exp-diff.east-high.3col.pdf', h=8.75, w=7.5, useDingbats = FALSE)
ggsave('plots/amp-exp-diff.east-high.3col.standard-size.png', h=6.35, w=7.5)
ggsave('plots/amp-exp-diff.east-high.3col.standard-size.pdf', h=6.35, w=7.5, useDingbats = FALSE)

plot.ampdiff.summary()
```





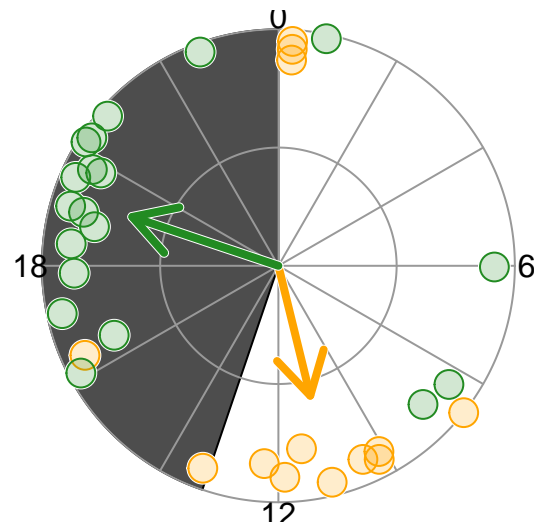
```
# ggsave("plots/amp-exp-diff-summary.png", w=5, h=7)
write.table(subset(cosopt, GeneID %in% west.high), 'cosopt-processed/cosopt-processed.amp-exp-diff.west.txt')
write.table(subset(cosopt, GeneID %in% east.high), 'cosopt-processed/cosopt-processed.amp-exp-diff.east.txt')

# 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, useDingbats = FALSE)
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
```

```

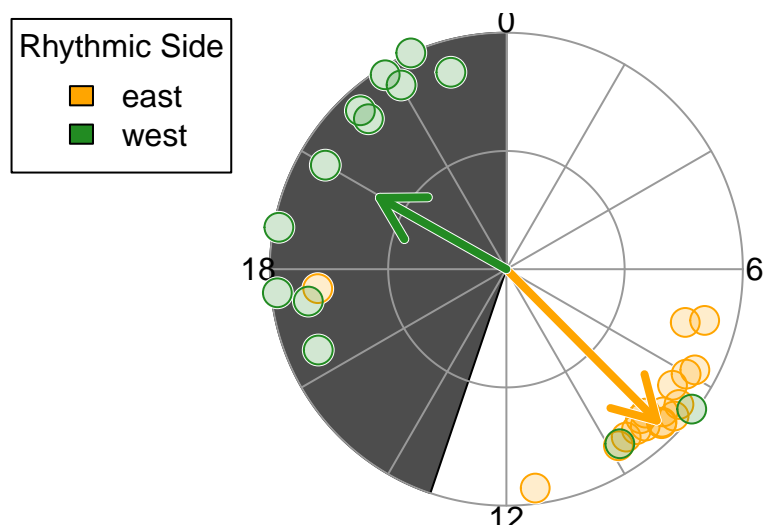
east.phase <- subset(cosopt, GeneID %in% east.rhythmic)$PeakPhase.E
west.phase <- subset(cosopt, GeneID %in% west.rhythmic)$PeakPhase.W

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

```

```

## pdf

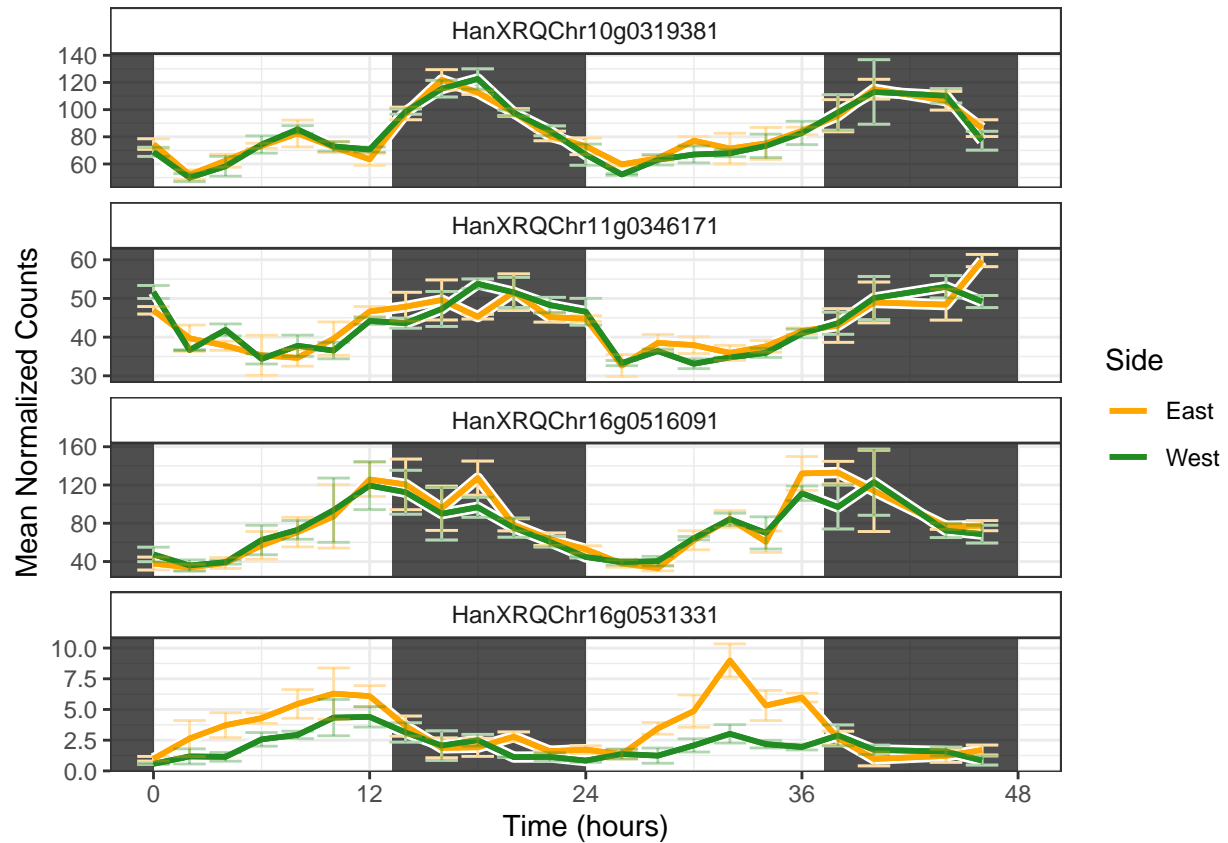
```

```
## 2
```

## Plotting GWAS Candidates

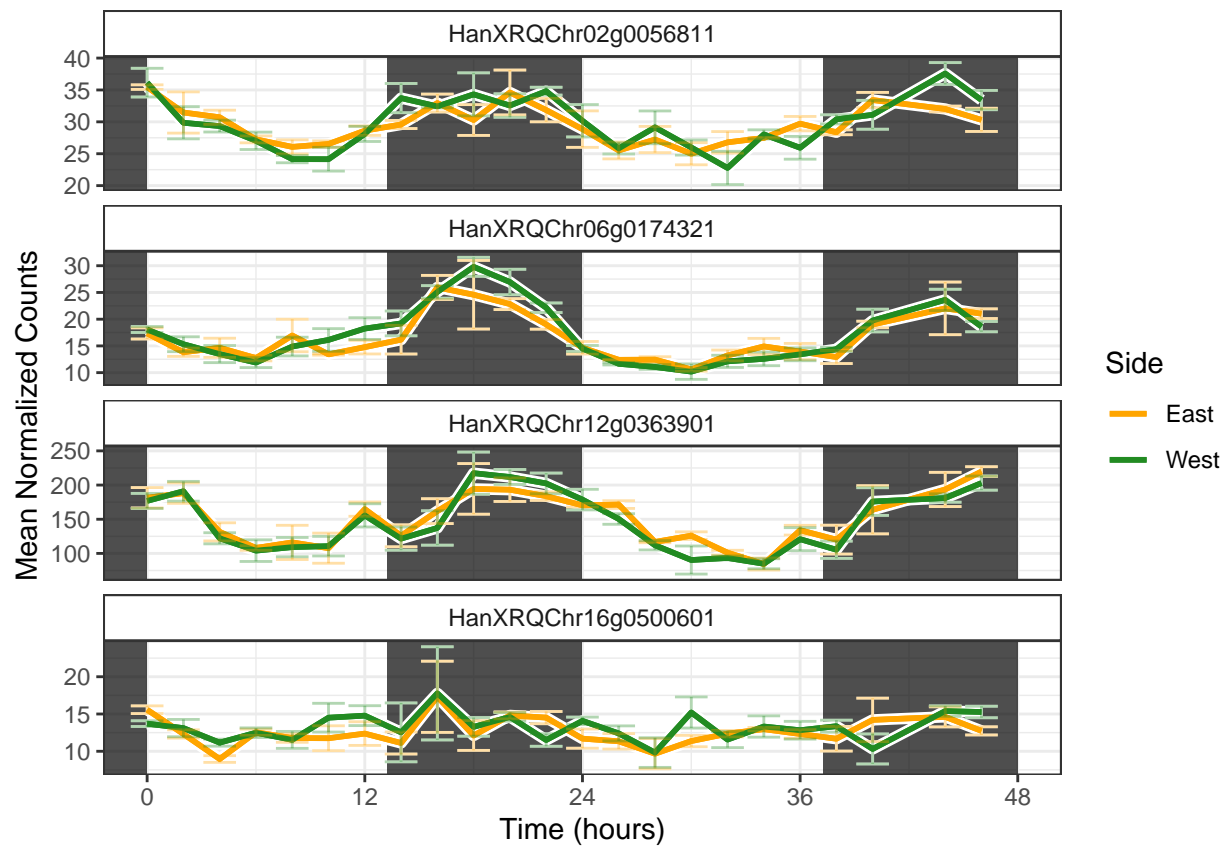
```
onset.time <- c('HanXRQChr10g0319381', 'HanXRQChr16g0516091', 'HanXRQChr16g0531331', 'HanXRQChr11g0346171')
nocturnal.reorientation <- c('HanXRQChr02g0056811', 'HanXRQChr16g0500601', 'HanXRQChr12g0363901', 'HanXRQChr16g0531331')
shoot.movement.pc1 <- c('HanXRQChr08g0210081', 'HanXRQChr03g0091141',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

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



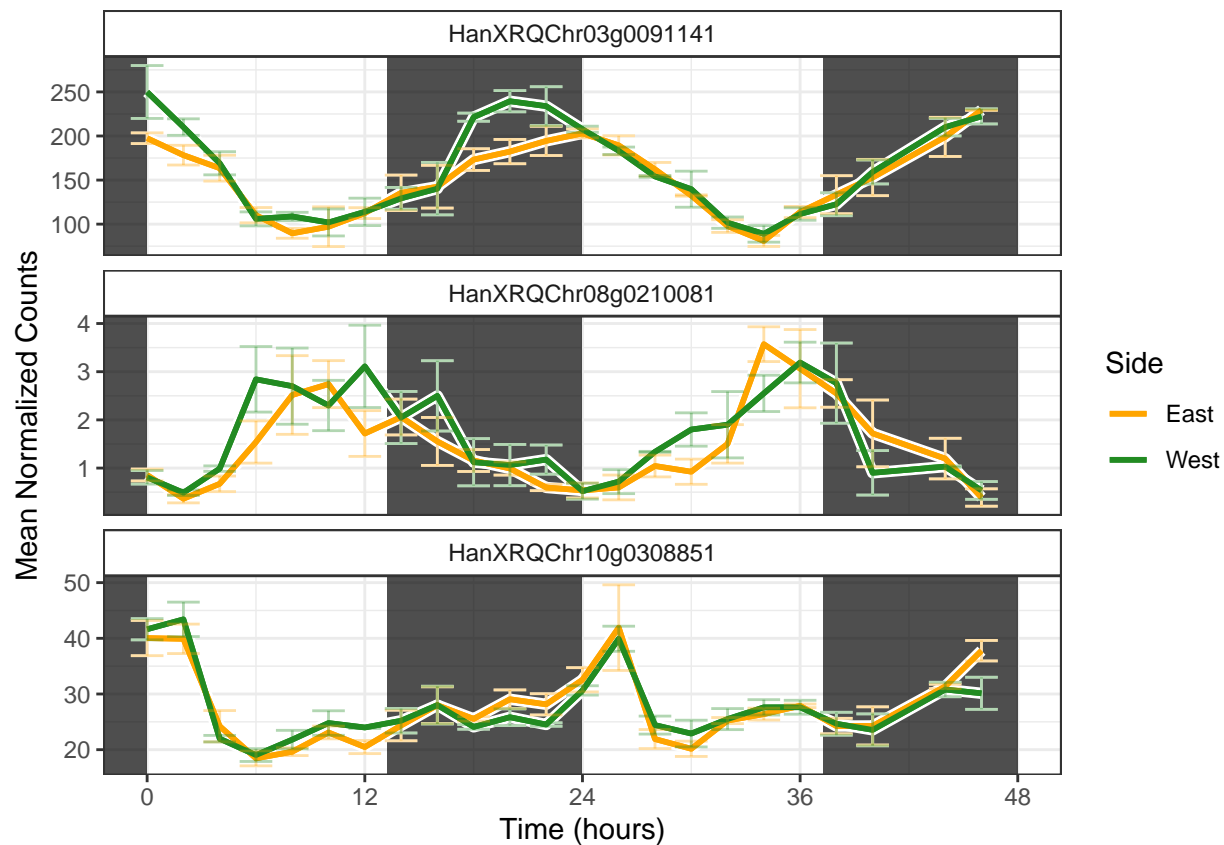
```
ggsave('plots/gwas.onset-time.png', w=4, h=6)
ggsave('plots/gwas.onset-time.pdf', w=4, h=6, useDingbats = FALSE)

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, useDingbats = FALSE)

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



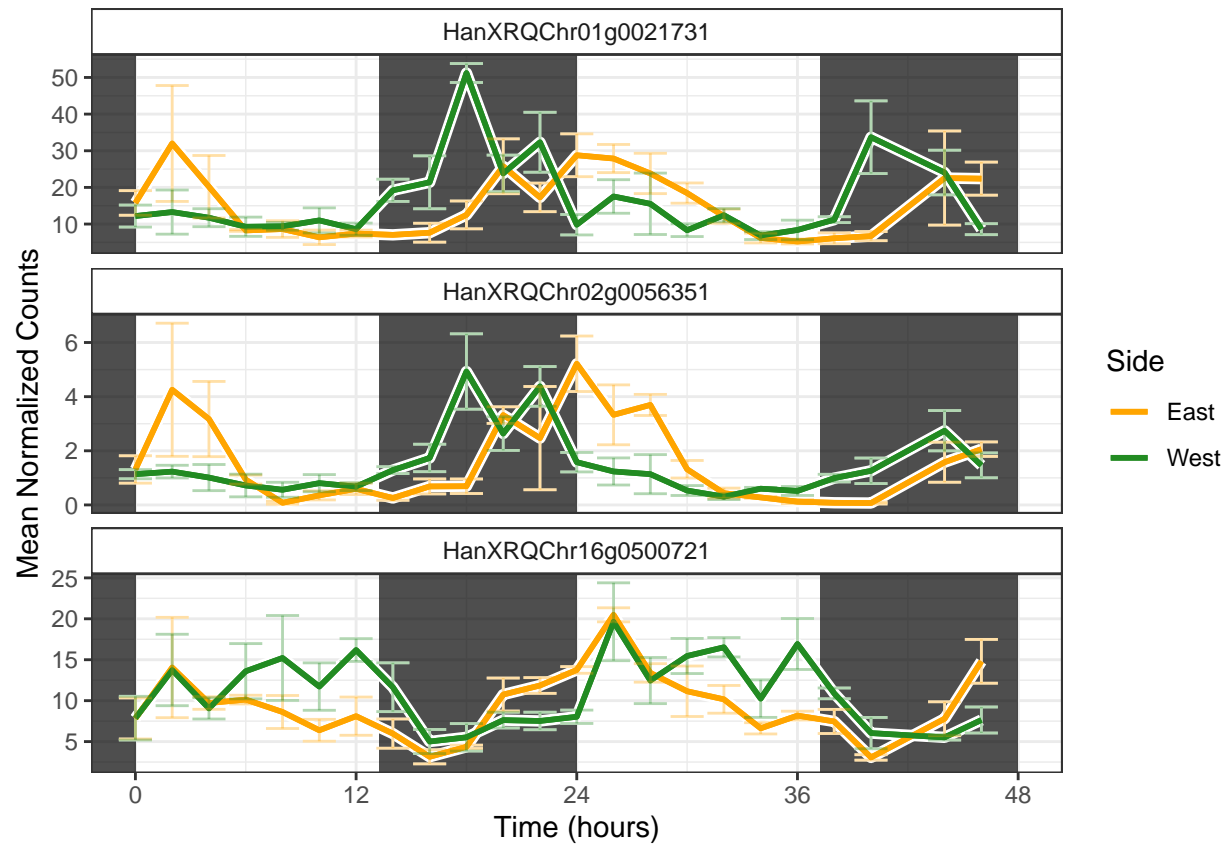
```
ggsave('plots/gwas.shoot-movement-pc1.png', w=4, h=4.7)
ggsave('plots/gwas.shoot-movement-pc1.pdf', w=4, h=4.7, useDingbats = FALSE)
```

```
# 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)
```

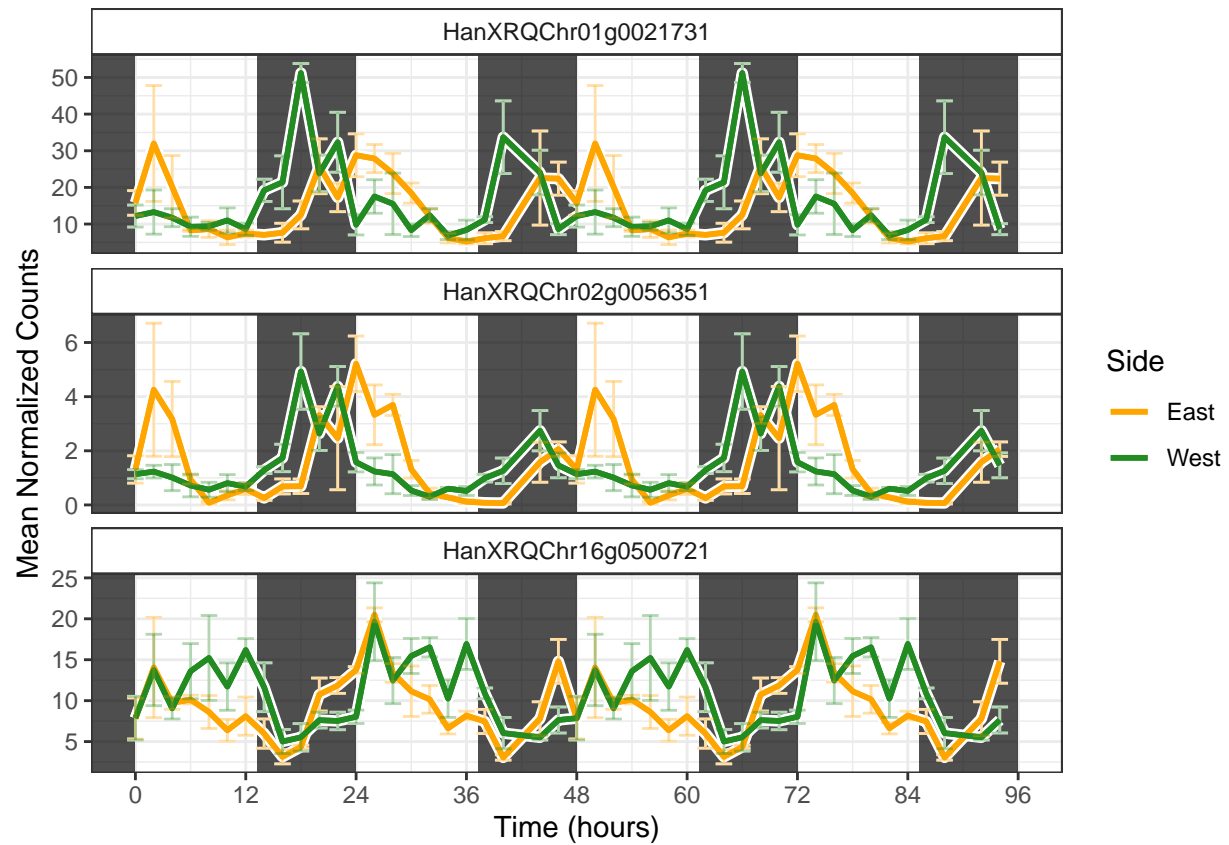
```
phase.shifted.genes <- c('HanXRQChr01g0021731', 'HanXRQChr02g0056351', 'HanXRQChr16g0500721')
```

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



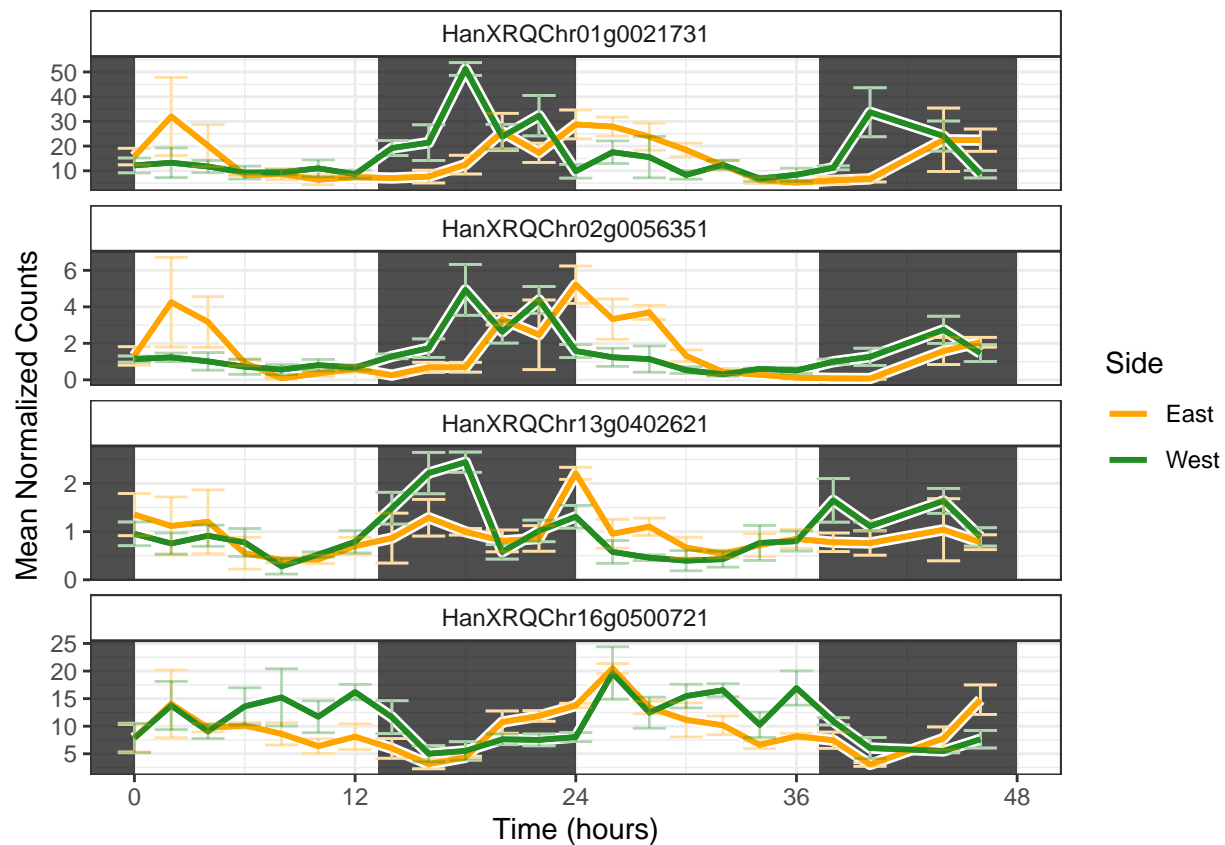
```
ggsave('plots/phase-shifted.png', w=4, h=4.7)
ggsave('plots/phase-shifted.pdf', w=4, h=4.7, useDingbats = FALSE)

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, useDingbats = FALSE)

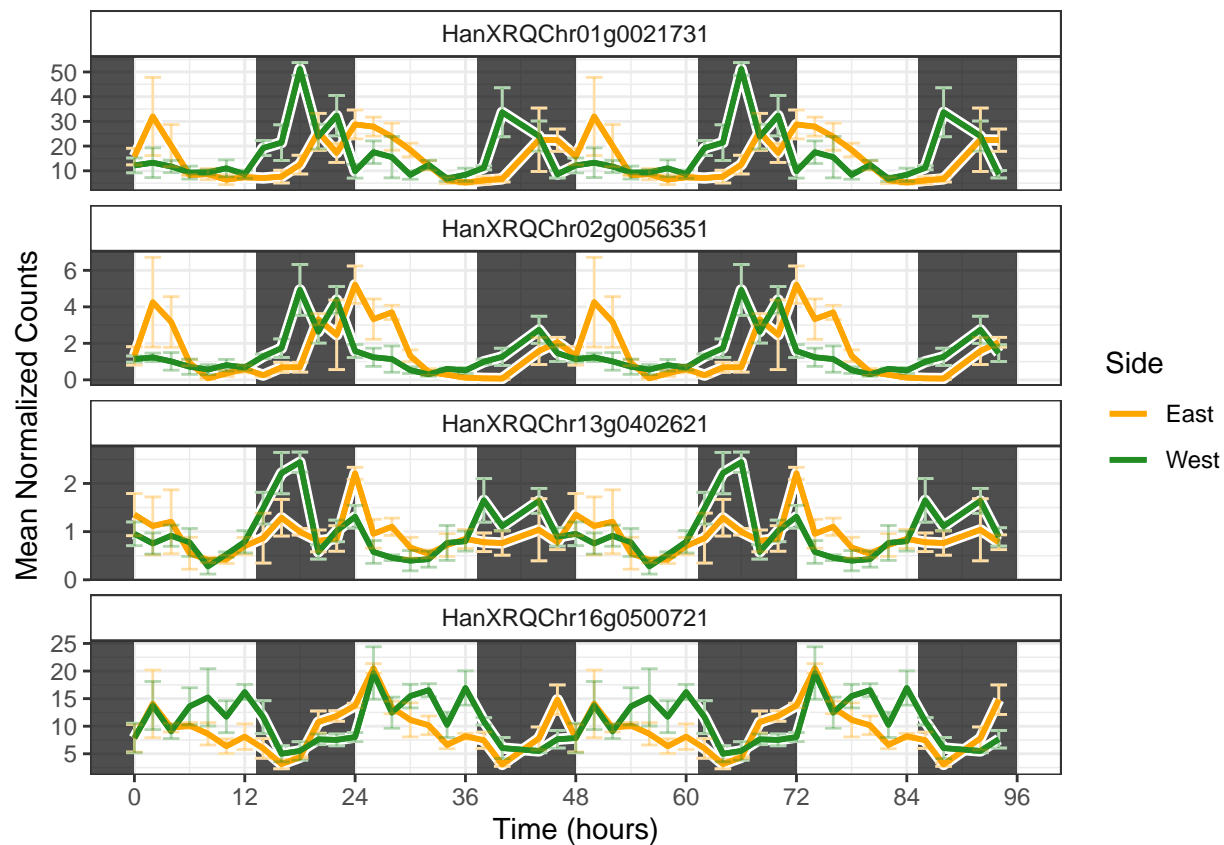
plot.timecourse(c(phase.shifted.genes, 'HanXRQChr13g0402621'), lights.off = 13.25)
```



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

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





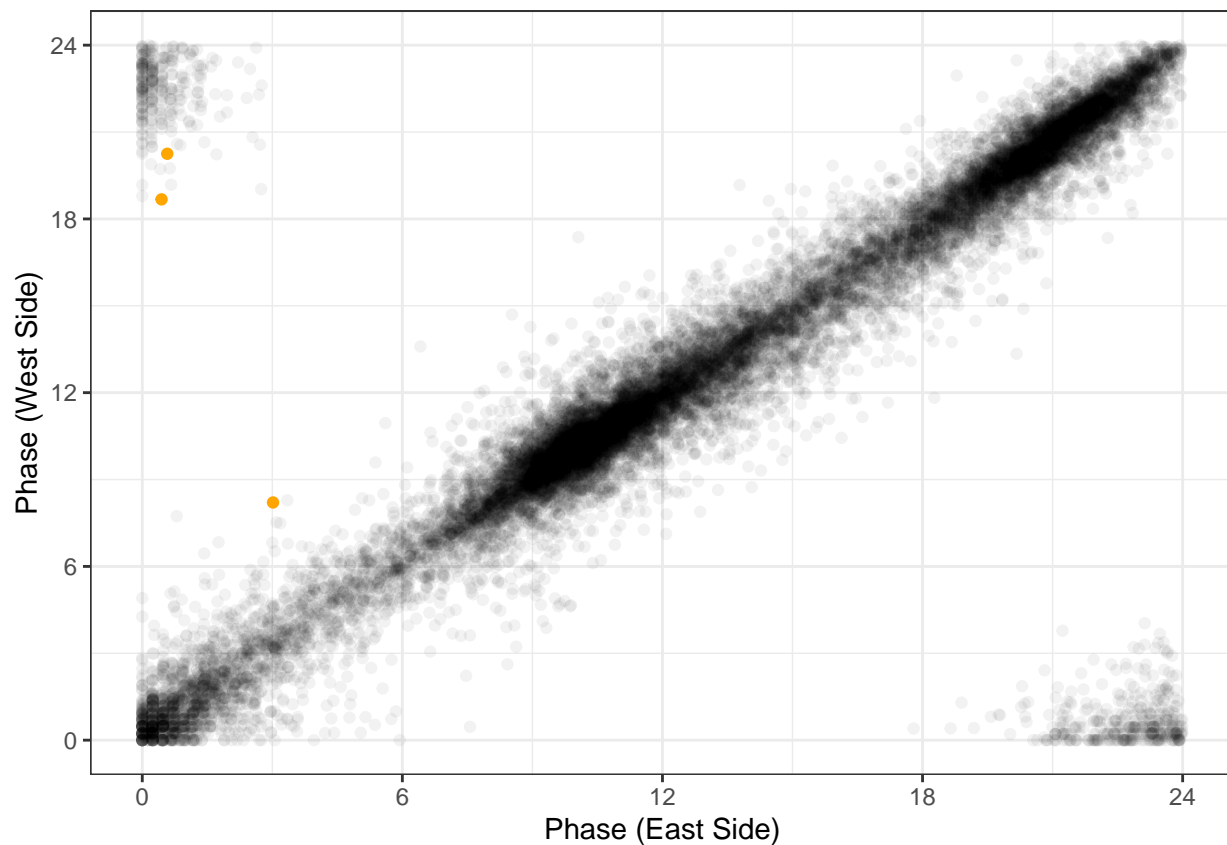
```

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, useDingbats = FALSE)

phase.shifted.color <- 'orange'

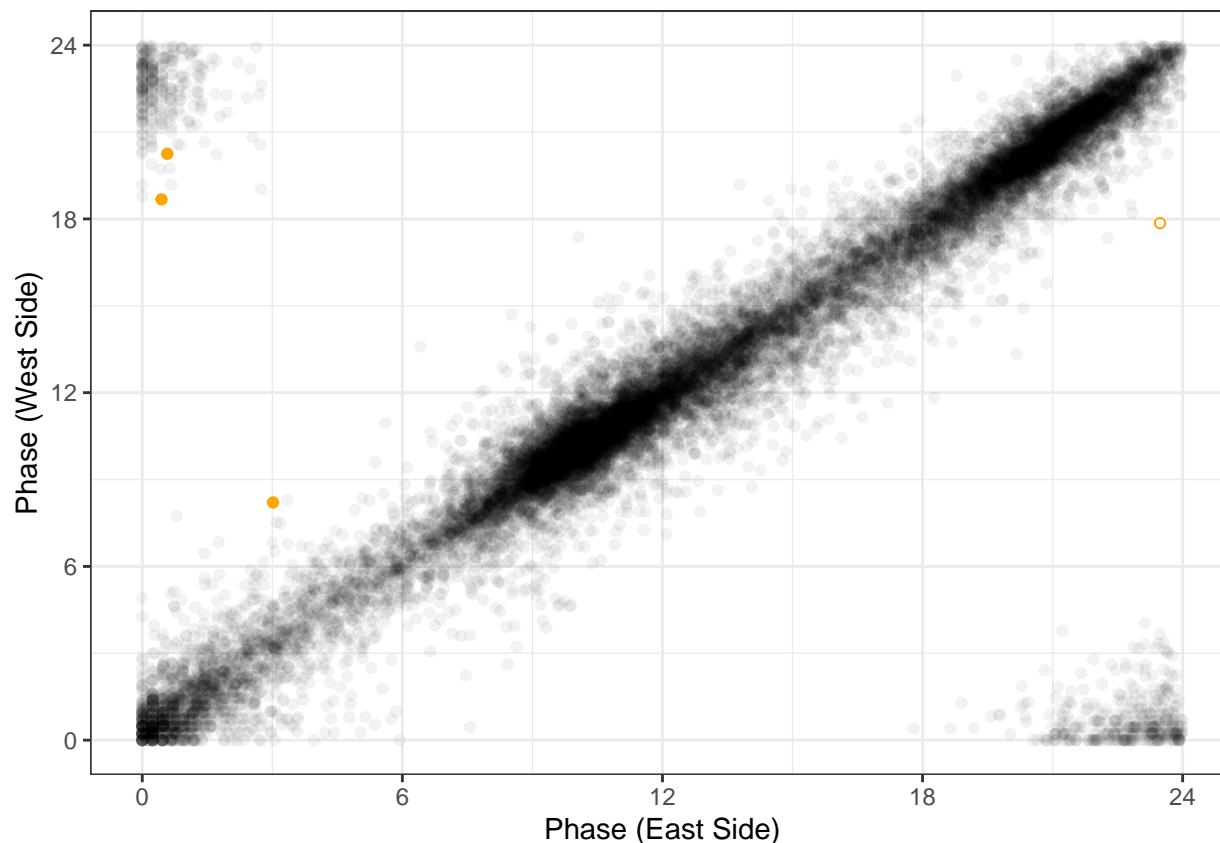
cosopt.both.phaseshifted <- subset(cosopt.both, GeneID %in% phase.shifted.genes)
ggplot(cosopt.both) +
  geom_point(aes(x = PeakPhase.E, y = PeakPhase.W), alpha=0.05) +
  geom_point(data = subset(cosopt, GeneID %in% phase.shifted.genes), aes(x = PeakPhase.E, y = PeakPhase
  scale_x_continuous(breaks=seq(0, 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, useDingbats = FALSE)

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, useDingbats = FALSE)
```

Create Summary Table with Time Course Data, COSOPT results, etc.

```
# Merge time course data with COSOPT results
timecourse.cosopt.summary <- merge(timecourse.summary, cosopt, by = 'GeneID', all = TRUE)

# Record mean expression levels
timecourse.cosopt.summary$MeanExpressionEast <- mean.expression$east
timecourse.cosopt.summary$MeanExpressionWest <- mean.expression$west
timecourse.cosopt.summary$MeanExpressionMerged <- mean.expression$merged

# Mark rhythmic genes
timecourse.cosopt.summary$RhythmicEast[timecourse.cosopt.summary$GeneID %in% cosopt$GeneID] <- 0
timecourse.cosopt.summary$RhythmicWest[timecourse.cosopt.summary$GeneID %in% cosopt$GeneID] <- 0
timecourse.cosopt.summary$RhythmicBoth[timecourse.cosopt.summary$GeneID %in% cosopt$GeneID] <- 0
timecourse.cosopt.summary$RhythmicMerged[timecourse.cosopt.summary$GeneID %in% cosopt$GeneID] <- 0

timecourse.cosopt.summary$RhythmicEast[timecourse.cosopt.summary$GeneID %in% rhythmic.east] <- 1
timecourse.cosopt.summary$RhythmicWest[timecourse.cosopt.summary$GeneID %in% rhythmic.west] <- 1
timecourse.cosopt.summary$RhythmicBoth[timecourse.cosopt.summary$GeneID %in% rhythmic.both] <- 1
timecourse.cosopt.summary$RhythmicMerged[timecourse.cosopt.summary$GeneID %in% rhythmic.merged] <- 1

# Mark expressed genes
```

```

timecourse.cosopt.summary$ExpressedEast[timecourse.cosopt.summary$GeneID %in% cosopt$GeneID] <- 0
timecourse.cosopt.summary$ExpressedWest[timecourse.cosopt.summary$GeneID %in% cosopt$GeneID] <- 0
timecourse.cosopt.summary$ExpressedBoth[timecourse.cosopt.summary$GeneID %in% cosopt$GeneID] <- 0
timecourse.cosopt.summary$ExpressedMerged[timecourse.cosopt.summary$GeneID %in% cosopt$GeneID] <- 0

timecourse.cosopt.summary$ExpressedEast[timecourse.cosopt.summary$GeneID %in% rownames(expressed)[expressed]] <- 0
timecourse.cosopt.summary$ExpressedWest[timecourse.cosopt.summary$GeneID %in% rownames(expressed)[expressed]] <- 0
timecourse.cosopt.summary$ExpressedBoth[timecourse.cosopt.summary$GeneID %in% rownames(expressed)[expressed]] <- 0
timecourse.cosopt.summary$ExpressedMerged[timecourse.cosopt.summary$GeneID %in% rownames(expressed)[expressed]] <- 0

# Mark genes with higher amplitude or expression on one side
timecourse.cosopt.summary$AmpHigherEast[timecourse.cosopt.summary$GeneID %in% rhythmic.both] <- 0
timecourse.cosopt.summary$AmpHigherWest[timecourse.cosopt.summary$GeneID %in% rhythmic.both] <- 0
timecourse.cosopt.summary$AmpHigherEast[timecourse.cosopt.summary$GeneID %in% amp.e] <- 1
timecourse.cosopt.summary$AmpHigherWest[timecourse.cosopt.summary$GeneID %in% amp.w] <- 1

timecourse.cosopt.summary$ExpHigherEast[timecourse.cosopt.summary$GeneID %in% rhythmic.both] <- 0
timecourse.cosopt.summary$ExpHigherWest[timecourse.cosopt.summary$GeneID %in% rhythmic.both] <- 0
timecourse.cosopt.summary$ExpHigherEast[timecourse.cosopt.summary$GeneID %in% exp.e] <- 1
timecourse.cosopt.summary$ExpHigherWest[timecourse.cosopt.summary$GeneID %in% exp.w] <- 1

timecourse.cosopt.summary$AmpExpHigherEast[timecourse.cosopt.summary$GeneID %in% rhythmic.both] <- 0
timecourse.cosopt.summary$AmpExpHigherWest[timecourse.cosopt.summary$GeneID %in% rhythmic.both] <- 0
timecourse.cosopt.summary$AmpExpHigherEast[timecourse.cosopt.summary$GeneID %in% amp.e | timecourse.cosopt.summary$AmpHigherEast] <- 1
timecourse.cosopt.summary$AmpExpHigherWest[timecourse.cosopt.summary$GeneID %in% amp.w | timecourse.cosopt.summary$AmpHigherWest] <- 1

# Mark asymmetric cyclers (rhythmic on one side, but not the other)
timecourse.cosopt.summary$AsymmetricEast[timecourse.cosopt.summary$GeneID %in% union(rhythmic.east, rhythmic.west)] <- 1
timecourse.cosopt.summary$AsymmetricWest[timecourse.cosopt.summary$GeneID %in% union(rhythmic.east, rhythmic.west)] <- 1
timecourse.cosopt.summary$AsymmetricEast[timecourse.cosopt.summary$GeneID %in% east.rhythmic] <- 1
timecourse.cosopt.summary$AsymmetricWest[timecourse.cosopt.summary$GeneID %in% west.rhythmic] <- 1

head(timecourse.cosopt.summary, n=5)

##           GeneID East_0h_mean East_2h_mean East_4h_mean East_6h_mean
## 1 HanXRQChr00c0001g0570931    0.9117024    0.45926241    0.4204598    0.2750039
## 2 HanXRQChr00c0003g0570971    0.1091173    0.07930041    0.3039131    0.2753755
## 3 HanXRQChr00c0003g0570981    0.4958849    0.40353149    0.5952826    0.4400805
## 4 HanXRQChr00c0004g0571001    5.6060333    4.89470893    5.5596748    5.6166944
## 5 HanXRQChr00c0004g0571011   11.1588551   15.52162866   22.0319606   21.7999671
##      East_8h_mean East_10h_mean East_12h_mean East_14h_mean East_16h_mean
## 1      0.4139209    0.50698195    0.4002296    0.2570277    0.7064449
## 2      0.2208127    0.06060066    0.2565281    0.1935906    0.2107729
## 3      0.6646679    0.49844351    0.5141270    0.6486378    0.7907490
## 4      6.2860724    6.26084198    6.9709343    8.8690336    9.9291800
## 5     25.3388315   24.38435423   19.6879723   14.2924269   15.3190341
##      East_18h_mean East_20h_mean East_22h_mean East_24h_mean East_26h_mean
## 1      0.3617501    0.7431298    0.1996955    0.2785472    0.5099242
## 2      0.2019911    0.1330436    0.2708863    0.1692589    0.1748185
## 3      0.6243119    0.7146057    0.5725981    0.8922424    0.9126925
## 4      9.0086614    9.7163352    6.6995663    5.6517962    6.6900929
## 5     16.0487244   13.5879497   17.5509128   15.1843145   20.3450989

```

##	East_28h_mean	East_30h_mean	East_32h_mean	East_34h_mean	East_36h_mean
## 1	0.48712893	0.9174001	0.2645752	0.8335053	0.9937369
## 2	0.09308733	0.0000000	0.2154461	0.2072242	0.1931378
## 3	0.86688849	0.4394522	0.6139451	0.4468905	0.5744472
## 4	5.69252070	5.4994014	4.5567243	6.7729709	6.2324946
## 5	20.83399118	21.1593221	22.4029475	23.3398537	25.2356634
##	East_38h_mean	East_40h_mean	East_44h_mean	East_46h_mean	West_0h_mean
## 1	0.8400430	0.5541328	0.7968898	0.2715544	0.3665759
## 2	0.2188712	0.3322028	0.2797127	0.2697980	0.1250066
## 3	0.8110953	0.8199693	1.1033019	0.7370958	0.8347580
## 4	9.4830179	8.6101784	9.9691223	7.5724658	6.2948094
## 5	19.2084475	13.3859965	18.6636127	18.1540090	13.2852110
##	West_2h_mean	West_4h_mean	West_6h_mean	West_8h_mean	West_10h_mean
## 1	0.6298716	0.8387638	0.4982465	0.2973809	0.8182397
## 2	0.2634719	0.1444436	0.1171651	0.2218034	0.1849692
## 3	0.4992268	0.6271347	0.2662168	0.8311336	0.7441114
## 4	4.3186672	6.6888407	6.0930987	7.6502346	7.2728022
## 5	16.6503097	20.2912081	21.7928923	25.9886879	26.6099122
##	West_12h_mean	West_14h_mean	West_16h_mean	West_18h_mean	West_20h_mean
## 1	0.4823134	0.2376923	0.6255419	0.3134011	0.3483086
## 2	0.1826446	0.2446365	0.3756466	0.1239564	0.1172849
## 3	0.7605737	0.5450553	0.7953738	1.3144125	0.3707426
## 4	7.0846240	9.6736792	7.2989631	10.1913688	10.5880762
## 5	22.0590107	17.2634986	14.8060528	17.9994852	15.9893356
##	West_22h_mean	West_24h_mean	West_26h_mean	West_28h_mean	West_30h_mean
## 1	0.2108630	0.3306077	0.4570837	0.42106947	1.0346923
## 2	0.3484867	0.3027098	0.2019638	0.06640825	0.1999967
## 3	0.7571385	0.7021352	0.7008402	0.69472634	0.9354931
## 4	6.7669680	5.7124289	5.7156136	6.09052839	7.3195046
## 5	18.3584199	13.6407907	18.3070608	16.35965598	25.2782878
##	West_32h_mean	West_34h_mean	West_36h_mean	West_38h_mean	West_40h_mean
## 1	0.3775487	0.3780953	1.0153553	0.8798077	0.4434399
## 2	0.2497154	0.4450860	0.4744737	0.2564431	0.2429383
## 3	0.6391529	0.5094483	1.0940143	0.9581114	0.8936439
## 4	5.8505074	6.6773925	7.7865504	10.0947371	8.6545831
## 5	23.1909166	24.4258443	23.4340058	17.5779978	12.8039267
##	West_44h_mean	West_46h_mean	Merged_0h_mean	Merged_2h_mean	Merged_4h_mean
## 1	0.6449330	0.6250404	0.6391391	0.5445670	0.6296118
## 2	0.2163135	0.3600761	0.1170620	0.1713862	0.2241784
## 3	0.9005424	0.9208543	0.6653214	0.4513792	0.6112087
## 4	7.7586798	7.0282456	5.9504214	4.6066881	6.1242577
## 5	18.0462215	17.5121522	12.2220331	16.0859692	21.1615843
##	Merged_6h_mean	Merged_8h_mean	Merged_10h_mean	Merged_12h_mean	Merged_14h_mean
## 1	0.3780742	0.3556509	0.6626108	0.4412715	0.2473600
## 2	0.2255616	0.2213080	0.1227849	0.2195863	0.2191135
## 3	0.3864247	0.7479007	0.6212775	0.6373503	0.5968465
## 4	4.9821429	6.9681535	6.7668221	7.0277791	9.2713564
## 5	21.8535890	25.6637597	25.4971332	20.8734915	15.7779628
##	Merged_16h_mean	Merged_18h_mean	Merged_20h_mean	Merged_22h_mean	
## 1	0.6659934	0.3375756	0.5457192	0.2052792	
## 2	0.2932098	0.1629737	0.1251643	0.3096865	
## 3	0.7930614	0.9693622	0.5426742	0.6648683	
## 4	8.6140715	9.6000151	10.1522057	6.7332671	
## 5	15.0625434	17.0241048	14.7886427	17.9546664	

##	Merged_24h_mean	Merged_26h_mean	Merged_28h_mean	Merged_30h_mean		
## 1	0.3045775	0.4835040	0.4575030	0.63973595		
## 2	0.2359843	0.1883911	0.1030196	0.07066642		
## 3	0.7971888	0.8067663	0.8317997	0.60402152		
## 4	5.6821125	6.2028532	5.6336810	5.46274041		
## 5	14.4125526	19.3260799	18.2888865	20.97061753		
##	Merged_32h_mean	Merged_34h_mean	Merged_36h_mean	Merged_38h_mean		
## 1	0.3210619	0.6058003	1.0045461	0.8599254		
## 2	0.2325807	0.3261551	0.3338057	0.2376571		
## 3	0.6265490	0.4781694	0.8342308	0.8846033		
## 4	5.2036159	6.7251817	7.0095225	9.7888775		
## 5	22.7969320	23.8828490	24.3348346	18.3932227		
##	Merged_40h_mean	Merged_44h_mean	Merged_46h_mean	East_0h_stderr	East_2h_stderr	
## 1	0.4987863	0.7209114	0.4482974	0.51888536	0.22023767	
## 2	0.2875706	0.2480131	0.3149371	0.05774153	0.04006672	
## 3	0.8568066	1.0019221	0.8289750	0.24316950	0.09346886	
## 4	8.6323808	8.8639011	7.3003557	0.48788290	0.53699144	
## 5	13.0949616	18.3549171	17.8330806	0.80726177	0.89031010	
##	East_4h_stderr	East_6h_stderr	East_8h_stderr	East_10h_stderr	East_12h_stderr	
## 1	0.21982503	0.16470500	0.12585793	0.15267188	0.2368528	
## 2	0.05232181	0.05552109	0.01906768	0.03056394	0.1017951	
## 3	0.10617768	0.21948279	0.22158508	0.20177774	0.2032670	
## 4	0.85571722	0.56004232	0.69126780	0.61560685	0.3291396	
## 5	4.51695765	1.39467615	2.51144988	5.71976722	1.4670510	
##	East_14h_stderr	East_16h_stderr	East_18h_stderr	East_20h_stderr		
## 1	0.07466866	0.2883887	0.2772860	0.28226508		
## 2	0.04669649	0.1064282	0.1175269	0.02559927		
## 3	0.19091907	0.2982260	0.3047939	0.28249898		
## 4	2.04777763	1.9125293	1.3224221	1.52700349		
## 5	2.29847573	1.3051259	1.3508942	1.26038922		
##	East_22h_stderr	East_24h_stderr	East_26h_stderr	East_28h_stderr		
## 1	0.04679541	0.03368115	0.07229755	0.07052773		
## 2	0.15879990	0.06782794	0.12021884	0.09308733		
## 3	0.15378347	0.10616874	0.20294814	0.13311444		
## 4	1.63673522	1.14270815	0.66714847	0.80959695		
## 5	2.30431472	1.96015747	1.63198142	0.74515121		
##	East_30h_stderr	East_32h_stderr	East_34h_stderr	East_36h_stderr		
## 1	0.18747917	0.14128371	0.28242765	0.348733261		
## 2	0.00000000	0.07403526	0.02366426	0.008979767		
## 3	0.10798845	0.09532817	0.03075830	0.146235088		
## 4	0.02499476	0.47049483	0.17526696	0.210853406		
## 5	0.82954440	2.13929396	1.06118683	1.651712156		
##	East_38h_stderr	East_40h_stderr	East_44h_stderr	East_46h_stderr		
## 1	0.2335623	0.16804911	0.14710853	0.13578561		
## 2	0.1682888	0.05006363	0.07065581	0.01375465		
## 3	0.3244117	0.12791362	0.16605026	0.07207517		
## 4	1.9511321	0.66780981	0.90504328	0.79362952		
## 5	1.2445824	0.52425397	1.03267652	0.61229346		
##	West_0h_stderr	West_2h_stderr	West_4h_stderr	West_6h_stderr	West_8h_stderr	
## 1	0.11027881	0.26599001	0.51539951	0.02756896	0.1062568	
## 2	0.06446473	0.01948796	0.09509699	0.06434489	0.1689268	
## 3	0.14939189	0.05378165	0.21479357	0.20332459	0.1362975	
## 4	1.14111653	0.60346719	1.17659033	1.83433471	0.7469721	
## 5	1.78600746	1.21480736	3.07656292	1.12658237	2.7555534	

##	West_10h_stderr	West_12h_stderr	West_14h_stderr	West_16h_stderr
## 1	0.16451441	0.089218243	0.04326365	0.24734057
## 2	0.05049299	0.007963469	0.13605317	0.06425125
## 3	0.24059398	0.152134104	0.29140678	0.10704522
## 4	0.53287244	0.457263046	0.93571941	1.91132827
## 5	3.40316164	2.746555302	1.36102900	0.89988803
##	West_18h_stderr	West_20h_stderr	West_22h_stderr	West_24h_stderr
## 1	0.06548834	0.13310020	0.05254025	0.1837088
## 2	0.12395638	0.05925089	0.07142887	0.1305290
## 3	0.07484874	0.02234413	0.08988316	0.1371072
## 4	0.59283627	0.96947280	0.90857071	0.5796440
## 5	1.83353566	0.66165505	0.78459800	1.8709014
##	West_26h_stderr	West_28h_stderr	West_30h_stderr	West_32h_stderr
## 1	0.07718986	0.11019653	0.67585358	0.11119440
## 2	0.12963690	0.06640825	0.05952557	0.02244022
## 3	0.16461992	0.03064384	0.16754889	0.05208067
## 4	0.87324153	0.64505189	2.00420819	0.53808176
## 5	1.70151966	1.43775503	4.62596875	0.92747972
##	West_34h_stderr	West_36h_stderr	West_38h_stderr	West_40h_stderr
## 1	0.05840804	0.2947063	0.2493757	0.1668322
## 2	0.23083118	0.2877115	0.1057607	0.1458297
## 3	0.08103739	0.2479449	0.1087722	0.4669611
## 4	0.20871195	1.0072399	1.2132324	2.7557216
## 5	1.28114344	2.0530750	0.9943625	0.5712625
##	West_44h_stderr	West_46h_stderr	Merged_0h_stderr	Merged_2h_stderr
## 1	0.43922303	0.1954358	0.27386622	0.17876656
## 2	0.07043235	0.1873304	0.05855184	0.02868705
## 3	0.09315458	0.2203679	0.16638355	0.04821517
## 4	0.79567345	1.0232390	0.33809296	0.18445752
## 5	0.80053181	1.1232488	1.28909205	0.17345952
##	Merged_4h_stderr	Merged_6h_stderr	Merged_8h_stderr	Merged_10h_stderr
## 1	0.36147349	0.063626208	0.08599453	0.09386122
## 2	0.07304869	0.004711376	0.07536094	0.01898482
## 3	0.14141596	0.276125836	0.13684390	0.16047801
## 4	1.01615327	0.208334218	0.53032099	0.57082109
## 5	3.50050083	1.667950930	2.55311064	3.94170639
##	Merged_12h_stderr	Merged_14h_stderr	Merged_16h_stderr	Merged_18h_stderr
## 1	0.09346653	0.02060531	0.25601364	0.1058988
## 2	0.04791591	0.08768712	0.02698078	0.1207417
## 3	0.17191487	0.19674193	0.16340663	0.1898213
## 4	0.38182002	1.48830424	1.63042502	0.9576292
## 5	1.85938847	1.76914034	1.08327013	0.2413207
##	Merged_20h_stderr	Merged_22h_stderr	Merged_24h_stderr	Merged_26h_stderr
## 1	0.10916340	0.009620635	0.08592177	0.009295351
## 2	0.01699613	0.078827033	0.05621290	0.066803213
## 3	0.13095457	0.069302457	0.08688610	0.104402286
## 4	1.12969886	1.258834972	0.66361956	0.645854401
## 5	0.93940172	1.530651682	1.84648199	1.565232131
##	Merged_28h_stderr	Merged_30h_stderr	Merged_32h_stderr	Merged_34h_stderr
## 1	0.005695347	0.150922206	0.11172473	0.13737028
## 2	0.103019620	0.008739765	0.03313899	0.12680534
## 3	0.058765471	0.041258726	0.06757415	0.02514776
## 4	0.863021574	0.556575451	0.48115114	0.18725476
## 5	1.082143198	1.356393067	1.26186672	1.12671118

##	Merged_36h_stderr	Merged_38h_stderr	Merged_40h_stderr	Merged_44h_stderr			
## 1	0.3020354	0.24055905	0.05189863	0.28880410			
## 2	0.1441086	0.07133654	0.04816188	0.06215200			
## 3	0.1953915	0.21629519	0.27694960	0.08517163			
## 4	0.4248268	1.47253904	1.57933733	0.44364553			
## 5	1.8194348	1.11854481	0.41624878	0.54279309			
##	Merged_46h_stderr	Beta	Beta.E	Beta.W	MeanExpLev	MeanExpLev.E	
## 1	0.03089423	NA	NA	NA	NA	NA	
## 2	0.09199941	0.034317	NA	NA	0.21501	NA	
## 3	0.14508214	0.088299	0.087048	0.09939	0.70022	0.66195	
## 4	0.50231607	1.758000	1.816700	1.60300	7.14470	7.08820	
## 5	0.73455879	4.366200	4.612000	4.64770	18.86900	18.85300	
##	MeanExpLev.W	PeakPhase	PeakPhase.E	PeakPhase.W	Period	Period.E	Period.W
## 1	NA	NA	NA	NA	NA	NA	NA
## 2	NA	12.375	NA	NA	27.5	NA	NA
## 3	0.7528	18.486	20.436	16.872	23.7	26.2	22.2
## 4	7.2759	16.281	16.683	15.844	24.3	24.9	23.3
## 5	19.0860	9.000	8.550	9.503	22.5	22.5	22.1
##	Phase	Phase.E	Phase.W	pMMC.Beta	pMMC.Beta.E	pMMC.Beta.W	RelAmp
## 1	NA	NA	NA	NA	NA	NA	NA
## 2	-12.375	NA	NA	0.87212000	NA	NA	0.1596065
## 3	5.214	5.764	5.328	0.53278000	0.51955000	0.8765800	0.1261018
## 4	8.019	8.217	7.456	0.00080078	0.00083223	0.0066876	0.2460565
## 5	-9.000	-8.550	-9.503	0.00056347	0.00048054	0.0017195	0.2313954
##	RelAmp.E	RelAmp.W	phase.diff	amp.diff	exp.diff.log2	MeanExpressionEast	
## 1	NA	NA	NA	NA	NA	0.5392629	
## 2	NA	NA	NA	NA	NA	0.1943256	
## 3	0.1315024	0.1320271	-3.564	0.0005247195	0.18554438	0.6600409	
## 4	0.2562992	0.2203164	-0.839	-0.0359828145	0.03770640	7.0499357	
## 5	0.2446295	0.2435136	0.953	-0.0011159318	0.01772067	18.8972119	
##	MeanExpressionWest	MeanExpressionMerged	RhythmicEast	RhythmicWest			
## 1	0.5336901	0.5216305	NA	NA			
## 2	0.2376365	0.2169911	0	0			
## 3	0.7519496	0.7060308	0	0			
## 4	7.3309088	7.1001045	1	1			
## 5	19.2030819	18.9414963	1	1			
##	RhythmicBoth	RhythmicMerged	ExpressedEast	ExpressedWest	ExpressedBoth		
## 1	NA	NA	1	1	1		
## 2	0	0	1	1	1		
## 3	0	0	1	1	1		
## 4	1	1	1	1	1		
## 5	1	1	1	1	1		
##	ExpressedMerged	AmpHigherEast	AmpHigherWest	ExpHigherEast	ExpHigherWest		
## 1	1	NA	NA	NA	NA		
## 2	1	NA	NA	NA	NA		
## 3	1	NA	NA	NA	NA		
## 4	1	0	0	0	0		
## 5	1	0	0	0	0		
##	AmpExpHigherEast	AmpExpHigherWest	AsymmetricEast	AsymmetricWest			
## 1	NA	NA	NA	NA			
## 2	NA	NA	NA	NA			
## 3	NA	NA	NA	NA			
## 4	0	0	0	0			
## 5	0	0	0	0			



```
write.table(timecourse.cosopt.summary, "Expression-and-COSOPT-Summary.txt", sep = "\t", quote = FALSE, c
```

```
session.info <- devtools::session_info()
session.info
```

```
## - Session info -----
## setting value
## version R version 4.0.1 (2020-06-06)
## os      macOS Catalina 10.15.4
## system  x86_64, darwin17.0
## ui      X11
## language (EN)
## collate en_US.UTF-8
## ctype   en_US.UTF-8
## tz      America/Los_Angeles
## date    2020-06-13
##
## - Packages -----
## package      * version      date      lib
## assertthat    0.2.1        2019-03-21 [1]
## backports     1.1.7        2020-05-13 [1]
## boot          1.3-25       2020-04-26 [1]
## callr         3.4.3        2020-03-28 [1]
## circular      * 0.4-93       2017-06-29 [1]
## cli           2.0.2        2020-02-28 [1]
## clockplot     * 0.0.0.9000  2020-06-13 [1]
## colorspace    1.4-1        2019-03-18 [1]
## crayon        1.3.4        2017-09-16 [1]
## desc          1.2.0        2018-05-01 [1]
## devtools      2.3.0        2020-04-10 [1]
## digest        0.6.25       2020-02-23 [1]
## dplyr         1.0.0        2020-05-29 [1]
## ellipsis      0.3.1        2020-05-15 [1]
## evaluate      0.14         2019-05-28 [1]
## fansi         0.4.1        2020-01-08 [1]
## farver        2.0.3        2020-01-16 [1]
## formatR       1.7          2019-06-11 [1]
## fs            1.4.1        2020-04-04 [1]
## futile.logger * 1.4.3        2016-07-10 [1]
## futile.options 1.0.1        2018-04-20 [1]
## generics      0.0.2        2018-11-29 [1]
## ggplot2       * 3.3.1        2020-05-28 [1]
## glue          1.4.1        2020-05-13 [1]
## gtable        0.3.0        2019-03-25 [1]
## htmltools     0.4.0        2019-10-04 [1]
## knitr         1.28         2020-02-06 [1]
## labeling      0.3          2014-08-23 [1]
## lambda.r      1.2.4        2019-09-18 [1]
## lifecycle     0.2.0        2020-03-06 [1]
## magrittr      1.5          2014-11-22 [1]
## memoise       1.1.0        2017-04-21 [1]
## munsell       0.5.0        2018-06-12 [1]
## mvtnorm       1.1-1        2020-06-09 [1]
## pillar        1.4.4        2020-05-05 [1]
```

[illegible]

