```r
library(caret)
library(ggbiplot)
library(dplyr)
library(GGally)


library(kernlab)
library(pls)

set.seed(1000)

training <- read.csv("~/Dropbox/Coursera/Predmachlearn_CP/pml-training.csv")
tr_classe <- training$classe

# Plot Predictors -------------

pcaplot <-function(x) {
        tr_pca <- prcomp(select(training, starts_with(x)),
                         center=TRUE,
                         scale.=TRUE)
        ggbiplot(tr_pca,
                 obs.scale = 1,
                 var.scale = 1,
                 groups = tr_classe,
                 ellipse = TRUE,
                 circle = TRUE) +
                scale_color_discrete(name = '') +
                theme(legend.direction = 'horizontal',
                      legend.position = 'top')

}

tr_nofactor <- dplyr::select(training19622, -c(classe, user_name))
trcor <- cor(tr_nofactor)

hm <- heatmap(trcor, Rowv = NA, Colv = NA, col=heat.colors(256), scale="column",
margins=c(5,10))

with(training11026,
     plot3d(gyros_dumbbell_x,
            gyros_dumbbell_y,
            gyros_dumbbell_z,
            type="s",
            col=as.numeric(classe)))

#pairs(training19622 %>% select(ends_with("_dumbbell")), col=training19622$classe)

box_facet <- function(df, c) {
        x <- dplyr::select(df, classe, contains(c))
        xtall <- gather(x, "meas","val",-classe)
        qplot(x=classe, y=val, data=xtall) + geom_boxplot() + facet_wrap(~meas, scales="free")
}
box_facet(training19619, "_x")
box_facet(training19619, "_y")
box_facet(training19619, "_z")
box_facet(training19619, "total_accel")


x <- dplyr::select(training19619, classe, user_name, contains("roll_"), contains("pitch_"),
contains("yaw_"))
xtall <- gather(x, "meas","val",-classe)
qplot(x=classe, y=val, data=xtall) + geom_boxplot() + facet_wrap(~meas, scales="free")

x <- dplyr::select(training19619, classe, user_name, contains("total_accel"))
```

```r
ggpairs(x,
        diag=list(continuous="density", discrete="bar"),
        upper=list(continuous="density", discrete="bar", mixed="box"),
        color="user_name")

# Remove outliers
training19619 <- training19622 %>% filter(accel_belt_x > -100 &
                                 gyros_dumbbell_x > -200 &
                                 gyros_forearm_x > -20 &
                                 gyros_dumbbell_y < 40 &
                                 magnet_dumbbell_y > -1000 &
                                 gyros_forearm_y < 300 &
                                 accel_forearm_y < 750 &
                                 gyros_dumbbell_z < 300 &
                                 gyros_forearm_z < 200 &
                                 total_accel_dumbbell < 50 &
                                 total_accel_forearm < 90)


### ---- NEED TO DO SOME TRANSFORMS ON THE DATA, SUCH AS TRAJECTORY, VELOCITY, ACCELERATION

nzv <- nearZeroVar(training19619)
names(training)[nzv]
# this was null.


dv <- dummyVars(~user_name, data=training)
dv <- data.frame(predict(dv, newdata=training), training19622)

# Simplest possible model
tr <- training19619 %>% filter(user_name=="carlitos")
trfit <- train(classe ~ user_name +
                       roll_dumbbell +
                       magnet_arm_y +
                       magnet_dumbbell_y +
                       magnet_forearm_y +
                       accel_belt_z +
                       magnet_belt_z,
               data=training19619,
               method="rf")
cm <- confusionMatrix(trfit)
heatmap(as.matrix(cm$table), Rowv=NA, Colv=NA)


# Train ------------------------
train_fit <- train(classe ~ .,
                   filter(training19619, user_name=="carlitos") %>% select(-user_name),
                   preProcess=c("knnImpute", "center", "scale"),
                   trainControl(method="repeatedcv"),
                   method="rpart")
confusionMatrix(train_fit)

#,
#starts_with("pitch"),
#starts_with("yaw")

# Select covariates ------------
training19622 <- training %>%
        dplyr::select(classe,
               user_name,
               gyros_belt_x,     accel_belt_x,     magnet_belt_x,
               gyros_belt_y,     accel_belt_y,     magnet_belt_y,
               gyros_belt_z,     accel_belt_z,     magnet_belt_z,
               gyros_arm_x,      accel_arm_x,      magnet_arm_x,
               gyros_arm_y,      accel_arm_y,      magnet_arm_y,
               gyros_arm_z,      accel_arm_z,      magnet_arm_z,
               gyros_dumbbell_x, accel_dumbbell_x, magnet_dumbbell_x,
               gyros_dumbbell_y, accel_dumbbell_y, magnet_dumbbell_y,
               gyros_dumbbell_z, accel_dumbbell_z, magnet_dumbbell_z,
               gyros_forearm_x,  accel_forearm_x,  magnet_forearm_x,
               gyros_forearm_y,  accel_forearm_y,  magnet_forearm_y,
               gyros_forearm_z,  accel_forearm_z,  magnet_forearm_z,
               total_accel_belt,
               total_accel_arm,
               total_accel_dumbbell,
               total_accel_forearm,
               roll_dumbbell,  roll_forearm,  roll_arm,
               pitch_dumbbell, pitch_forearm, pitch_arm,
               yaw_dumbbell,   yaw_forearm,   yaw_arm)

#testing <- read.csv("~Dropbox/Coursera/Predmachlearn_CP/pml-testing.csv")
#test_fit <- predict(train_fit, newdata=testing)
```