Analysis of Swimming Race Data Set

Xinan Wang

MSc. Statistical Science, MT 2020, Week 4

1 Introduction

The data set contains 446 rows of data, each recording the distance, gender, course, stroke and time taken by a swimmer of a swimming event. My goal is to investigate the effects of the other variables on the competitors' swimming race result. Equivalently speaking, I want to search for evidence to conclude whether the exploratory variables have an impact on the time of the swimmers. I aim to achieve this via fitting a linear model.

2 Data Exploration

The response variable time, as well as an explanatory variable dist, are of class numeric. Moreover, in the data set dist only takes 4 different values. Other variables are all categorical, which means that the data set can be split to different groups. One can also observe that, across the groups, the range and scale of time varies substantially. For instance, the races with stroke of freestyle tend to take less time than those breaststroke races, and male swimmers are mostly faster than their female peers. Therefore, I make exploratory plots across different groups, from which I aim to gain some insights into the data.

I here show two sets of boxplots based on the data of 50m and 400m races. The first column are based on the race times for the two genders. In 50m races, the upper quartile of time taken by male swimmers is slightly lower than the minimum of female swimmers, indicating a large gender difference. On the other hand, this difference is narrowed in 400m races, although the male swimmers still have a significant advantage. The middle column of time against course type looks interesting, since long course is related to shorter time in

50m races but longer time in 400m events. Finally, the plots in the third column also suggest that the race times also vary according to different strokes.

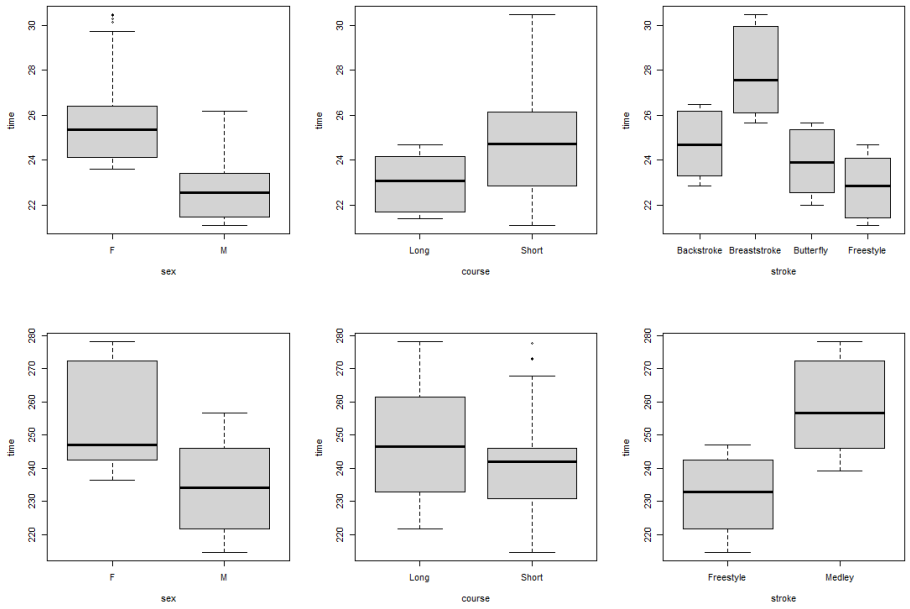

Fig. 1: Box plots of the race times based on different variables.
First Row: 50m races; **Second Row**: 400m races

Additionally, from the last column of Fig.1, it can be deduced that we do not have data for all strokes for races of some distances. To understand this fact more precisely, I produce the barplot displayed below to demonstrate how many races of each stroke are in the original data set for different distances.

In Fig.2, our data set is revealed to be unbalanced, i.e. the number of races for different strokes are not always equal, and there is a lack of some combinations of distances and strokes. For example, we do not have data for 400m backstroke, breaststroke and butterfly 400m races. However, it is still possible to make future predictions for these races, based on the data of the same strokes and different distances.

Fig. 2: Barplot of the race counts for different strokes and distances

3 Response Variable Transformation

This short section includes how I transformed the response variable time using Box-Cox transformation.

The plots shown in Fig.3 can be obtained by fitting the Box-Cox power transformation, where the model treats all variables as independent explanatory variables. Note that this choice of model is yet to be fixed, and I will test other choices in the following section.

The graph on the right shows clearly that $\lambda = 1$ does not fall into the 95% confidence interval, and thus the effect of transformation is significant. The MLE for λ in this case is ~ 0.895. This is logically reasonable, as the swimmers' speeds tend to be slower during long distance races, thus raising the dist to some power $\lambda < 1$ can handle this non-linearity between time and distance. Based on this, I transform the response before fitting models and conducting hypothesis testing analysis.

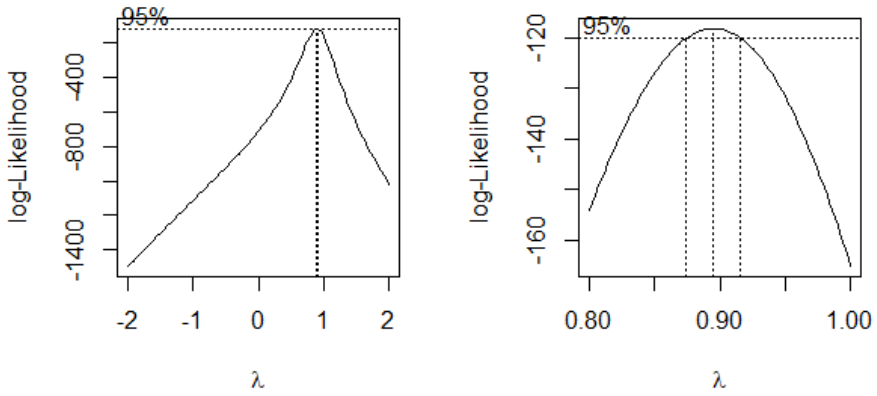

Fig. 3: Log-likelihood against choice of λ

4 Model Selection and Outlier Analysis

I now want to fit a linear model to find the underlying relation between time and other variables. A natural choice of model to start from has already been used in the last section, which involves all the other variables as independent explanatory variables. I want to search for evidence to: $^{(1)}$ exclude any variables, and $^{(2)}$ support the existence of the interaction terms. In addition, after the model has been fixed, I want to find potential outliers, which are the points having substantial effect on the model fit.

4.1 Variable Selection

Now start from the standard model including all variables. In the model summary, it can be observed that, all variables including all levels for categorical variables are significant (mostly with order $e-16$, except for the butterfly stroke. Specifically, the significance levels are obtained by conducting t-tests with $n-p$ degrees of freedom, where the null hypotheses are a specific level or variable is zero. This provides strong evidence that all individual variable needs to be included when all the others are involved.

I then perform F-tests to reassure that the variables are indeed necessary to be included. To achieve this, I fit nested models, each with one variable excluded. Subsequently, from the summaries of anova on the original model and each nested model, we can see large F-test statistics and tiny p-values. Therefore, I conclude that every variable in the data set is significant and model.

4.2 Adding Interaction

From the data exploration, we have observed that the categorical variables have different effect on the racing times for events of different lengths. Therefore, besides including the variables individually, I also conduct tests on whether the interactions should be added.

Start again from the model with all variables included independently, and add interaction terms between dist and each of the other categorical variables, as these interactions are revealed by the boxplots. After fitting these models and apply anova on each of them, the F statistics are all significantly large with negligible p-values. So interactions between distance and each of the other categorical variables are significant and I want to add them to my model.

Next, I fit several models adding more interactions among pairs of categorical variables. However, none of the resulting summaries show advantage over the model which do not involve categorical variable interaction, and hence I decide to stop here. To conclude, my final model involves all the variables, with additional interaction terms between dist and each of the other variables.

4.3 Detecting Potential Outliers

In order to visualise the model fit and search for potential outliers, I produce the diagnostic plots shown in Fig.4, where the points with large Cook's distances coloured red. The top left quantile plot suggests that the residuals are mostly t-distributed, except for those red points on the tail. Interestingly, the top right plot suggests that the variance of residuals grow as fitted values increase, which form four clusters, each containing samples with the same racing distance.

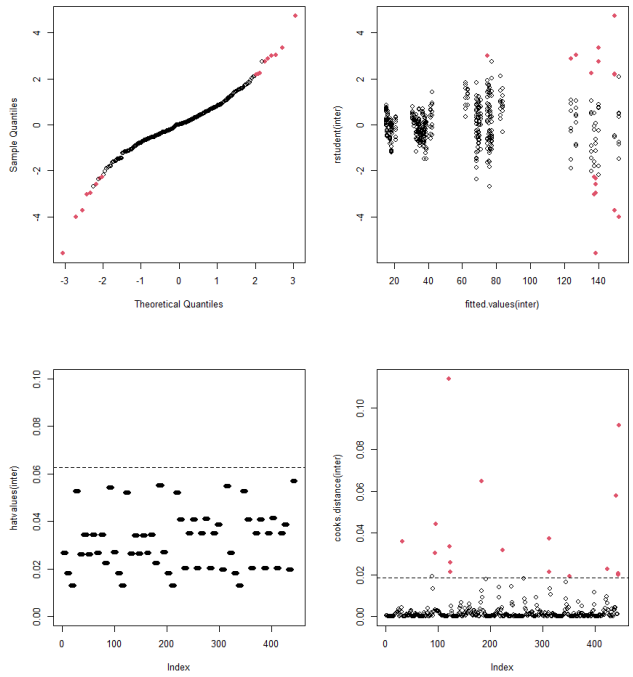

Fig. 4: Diagnostic Plot of My Choice of Model
Top Left: Studentised Residual and Theoretical t-Dist
Top Right: Studentised Residual against Fitted Values
Bottom Left: Leverages with $y = \frac{2p}{n}$ as the Dotted Line
Bottom Right: Cook's Distances with $y = \frac{8}{n-2p}$ Dotted

In the leverage plot, no points lie above the dotted line, while there are a number of points with relatively large Cook's distance. This indicates that in the current model, no sample affects the model fit very much, although some samples have much larger residuals. Furthermore, most red points belong to the 400m race cluster which has the largest variance. Therefore, directly deleting them at this stage is not a wise way to deal with this situation.

4.4 Weighted Regression

As the variances of the studentised residuals for races vary with different distances, I want to refit the same model but weight the samples, such that samples with higher residual variances are assigned

larger weights. The variances of residuals of 50m, 100m, 200m and 400m race events are $0.24, 0.25, 1, 06$ and 3.46 respectively, and fitting the weighted model produces the following diagnostics.

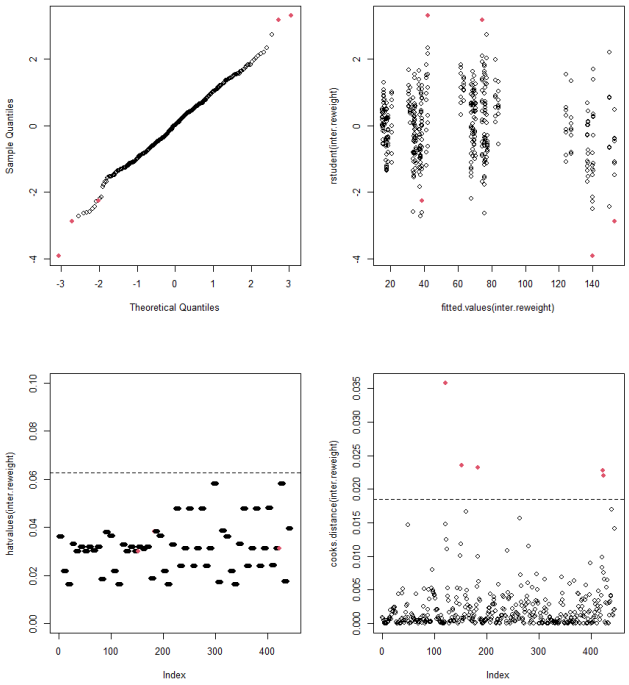

Fig. 5: Diagnostic Plot of My Choice of Model
With the Same Layout as Fig.4

Much fewer points are coloured red, so we have fewer candidates for outliers. The qqplot has much lighter tails, and the plot of studentised residuals against fitted values is substantially improved, with no heteroscedastic trend shown. Again, no point lie above the line in the leverage plot, although some of them are shown to have large Cook's distances due to high residuals.

4.5 Outlier Analysis

Now I want to consider the context of the data and analyse whether the high influence points are indeed outliers. It is known that time records one of the swimmers in the final in seconds. Here I print out the details of the samples which are coloured red in Fig.5, together with their studentised residuals and corresponding world records.

Index	Sex	Dist	Course	Stroke	Time	Record	Rstudent
121	F	400	Long	Freestyle	236.46	236.46	−3.91
152	F	100	Long	Breaststroke	68.10	64.13	+3.30
184	F	400	Long	Medley	266.36	266.36	−2.88
423	F	200	Short	Butterfly	129.42	119.61	+3.18
424	F	100	Short	Medley	57.24	56.51	−2.25

Table 1: Details of Potential Outlier Samples

From Table 1, notice that times of samples 121 and 184 are exactly equal to the world record, and others do not differ from their respective records too much. Even for sample 423, whose time is nearly 10 seconds longer than the world record, observing such a result is not unusual during a 200m butterfly short course race. For instance, in the final of the same event in 2019 World Championship, British swimmer Laura Stephens finished her race in 129.35 seconds and ranked at the 8^{th} place.

After the above analysis, together with the fact that there are no samples with large values of leverages, I conclude that the high influence points are not outliers, and I shall include them in my model.

5 Model Interpretation

Having fixed and fitter my model, I now move to the stage where I want to interpret it.

I firstly write down the mathematical expression of my model.

$$y = \beta_0 + \beta_1 d + \sum_{i=2}^{5} \gamma_i s_i + \alpha_2 g_2 + \eta_2 c_2 + \sum_{i=2}^{5} \delta_i d s_i + \phi_2 d g_2 + \psi_2 d c_2 + \epsilon$$

where y is the transformed response variable, time^{λ} (λ is just the Box-Cox MLE), β_0 is the intercept, $\beta_1 d$, $\alpha_2 g_2$, $\gamma_2 s_2$ and $\eta_2 c_2$ are the main effects of dist, sex, stroke and course respectively, and the other terms represent interaction terms between the two corresponding variables.

To interpret this model, it is necessary to specify levels of the categorical variables. If again write them out mathematically, we have

$$y = \begin{cases} \beta_0 + \beta_1 d + \epsilon, & \text{F, Backstroke, Long} \\ \beta_0 + \alpha_2 + (\beta_1 + \phi_2)d + \epsilon, & \text{M, Backstroke, Long} \\ \beta_0 + \gamma_2 + \eta_2 + (\beta_1 + \delta_2 + \psi_2)d + \epsilon, & \text{F, Breaststroke, Short} \\ \cdots \ \cdots \end{cases}$$

Knowing the complexity caused by the different levels, I will not interpret the model in all but the above three scenarios.

5.1 Case 1: Female Long Course Backstroke Races

This is the base case where all the categorical variables are at baseline. Our model is as simple as $y = \beta_0 + \beta_1 d + \epsilon$.

Read from the summary of the model, the estimates of the parameters are $\hat{\beta}_0 = 0.021, \hat{\beta}_1 = 0.381$. Therefore it can be interpreted in such a way: the time of female backstroke races in long course is modelled as a linear function of distance with noise, $f(d) = \beta_0 + \beta_1 d + \epsilon$, to the power of $\frac{1}{\lambda}$, where the function value is expected to increase by $\hat{\beta}_1$ as the distance of the swimming race increases by $1m$. The noise, meanwhile, is assumed to be normally distributed with zero mean, and its standard deviation has estimated to be 0.954. Due to the power, time is not a linear function of distance, but grows faster as the function takes larger values, i.e. as distance increases. The intercept here has no intrinsic meaning, as our data is not centered, and in swimming races the distance can never be 0.

5.2 Case 2: Male Long Course Backstroke Races

This case is the same as the previous one except for the main effect α_2 and the interaction between the M level of sex and the distance.

The model then becomes $y = \beta_0 + \alpha_2 + (\beta_1 + \phi_2)d + \epsilon$.

The estimates of the extra coefficients are $\hat{\alpha_2} = -0.557$ and $\hat{\phi_2} = -0.031$. This model can be interpreted as modelling the time of male backstroke races in long course varies as a linear function of distance to the power $\frac{1}{\lambda}$. The linear function has different coefficients with the one in last section, where the intercept becomes $\hat{\beta_0} + \hat{\alpha_2} = -0.536$ and coefficient of d is now $\hat{\beta_1} + \hat{\phi_2} = 0.350$. The expected race time is still not a linear function but grows faster with distance, while the rate becomes smaller due to a smaller coefficient of d.

5.3 Case 3: Female Short Course Breaststroke Races

This case has the most complex model form, which is $\beta_0 + \gamma_2 + \eta_2 + (\beta_1 + \delta_2 + \psi_2)d + \epsilon$, where the extra terms come from the short course, breast stroke and their interactions with distance.

The extra coefficients have estimated values $\hat{\gamma_2} = 0.320$, $\hat{\eta_2} = -0.064$, $\hat{\delta_2} = 0.037$ and $\hat{\psi_2} = -0.008$. This model has the same interpretation, again with different coefficients of the linear function. The intercept becomes $\hat{\beta_0} + \hat{\gamma_2} + \hat{eta_2} = 0.277$ and coefficient of d is $\hat{\beta_1} + \hat{\delta_2} + \hat{\psi_2} = 0.425$.

I also produce plots of the predicted times against distances for the above scenarios. The lines indeed curve slightly, indicating a power function ingredient, and the red dots representing the true data locate closely around the lines which is a signal of decent fit.

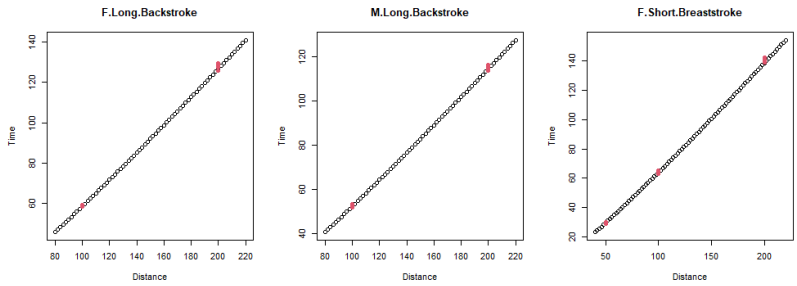

Fig. 6: Diagnostic Plot of Predicted Times by My Model

6 Predictions on the Given Races

In this very short section, I use my model to give out predicted times and prediction intervals for the four additional races.

race	Predicted Time	95% Prediction Interval
A	249.823	(245.877, 253.776)
B	26.949	(23.979, 29.953)
C	57.649	(54.437, 60.879)
D	60.431	(57.186, 63.694)

Table 2: Prediction Times and Intervals for New Races

The above results can also be computed by hand.

- The predicted times are $\hat{y} = (x^T\hat{\beta})^{\frac{1}{\lambda}}$, where x contains the variable values and levels and interactions, $\hat{\beta}$ is the estimated model parameter vector, and λ is the Box-Cox Transformation MLE.
- The prediction intervals are $\hat{y} \pm t_{433}(0.975)s\sqrt{1 + x^T(X^TX)^{-1}x}$, where $t_{433}(0.975)$ is the 97.5% quantile of a t-distribution of degrees of freedom $n-p = 433$, s is the residual standard deviance, and X is the design matrix.

7 Conclusion

My analysis starts from a full model which includes all the available variables independently. When searching for evidence for interaction terms among variables, I include interactions between dist and each of the other categorical variables. However, there is insufficient evidence to support that interactions among categorical variables exist. The model analysis is achieved by conducting t and F tests.

After my model is finalised, I fit it for the first time, produce diagnostic plots, reweight the samples, refit the model and analyse the context of the data. Then I conclude that the high influence points shall not be seen as outliers and deleted.

I interpret my model in three cases when the categorical variables take different values. The subsequent line plots illustrate that the model fit the data well. Finally, I give out prediction time and intervals for new races, with the formulae calculating them.

R Code

```
### Read Data
library("readr")
data <- read_csv("http://www.stats.ox.ac.uk/~laws/SB1/data/swim.csv")

### Relationship of time with different factors
par(mfrow=(c(2,3)))
pieces <- split(data,data$dist)
boxplot(time~sex, data=pieces[[1]])
boxplot(time~course,data=pieces[[1]])
boxplot(time~stroke,data=pieces[[1]])

boxplot(time~sex, data=pieces[[4]])
boxplot(time~course,data=pieces[[4]])
boxplot(time~stroke,data=pieces[[4]])

pieces2 <- split(data,data[,c('dist','stroke')])
count <- matrix(sapply(pieces2,nrow),ncol=5)
rownames(count) <- c('50m','100m','200m','400m')
colnames(count) <- c('Backstroke','Breaststroke','Butterfly',
                     'Freestyle','Medley')

## Check if the Data Set is Balanced
par(mfrow=c(1,1))

barplot(count,col=c('#606060','#808080','#A0A0A0','#C0C0C0','#E0E0E0'))
#title('Frequency of Different Stroke and Distance')
legend("topright",legend = c('50m','100m','200m','400m'),
       fill=c('#606060','#808080','#A0A0A0','#C0C0C0','#E0E0E0'))

### Box-Cox Transformation

par(mfrow=c(1,2))
library('MASS')
boxcox(time~dist+sex+stroke+course,data=data)
bc <- boxcox(time~dist+sex+stroke+course,data=data,
             lambda=seq(0.8,1,by=0.02))
lambda <- bc$x[which.max(bc$y)]

### Full Model and Test Droping Each Variable

full <- lm(I(time**lambda)~1+dist+stroke+sex+course,data=data)
summary(full)

full_stroke <- lm(I(time**lambda)~1+dist+sex+course,data=data)
anova(full,full_stroke)

full_course <- lm(I(time**lambda)~1+dist+sex+stroke,data=data)
```

```
anova(full,full_course)

full_sex <- lm(I(time**lambda)~1+dist+course+stroke,data=data)
anova(full,full_sex)

full_dist <- lm(I(time**lambda)~1+sex+course+stroke,data=data)
anova(full,full_dist)

### Involve Interaction

inter_course <- lm(I(time**lambda)~1+dist*course+stroke+sex,
                   data=data)
inter_stroke <- lm(I(time**lambda)~1+dist*stroke+sex+course,
                   data=data)
inter_sex <- lm(I(time**lambda)~1+dist*sex+stroke+course,
                data=data)
anova(full, inter_course)
anova(full, inter_stroke)
anova(full, inter_sex)

### Final Model
inter <- lm(I(time**lambda)~1+dist*(stroke+sex+course),data=data)

### Characteristics
p <- length(inter$coefficients)
n <- nrow(data)
# Large Cook's Distance
i <- cooks.distance(inter) > (8/(n - 2*p))

### Diagnostic Plots
par(mfrow=c(2,2))
# QQPlot
qqnorm(rstudent(inter), main = NULL, pch = 1 + 15*i, col = 1 + i)
qqline(rstudent(inter))

# Fitted Against Student Residual
plot(fitted.values(inter), rstudent(inter), pch = 1 + 15*i, col = 1 + i)

# Leverage
plot(hatvalues(inter), ylim=c(0,0.1), pch = 1 + 15*i, col = 1 + i)
abline(2*p/n, 0, lty = 2)

# Cook's Distance
plot(cooks.distance(inter), pch = 1 + 15*i, col = 1 + i)
abline(8/(n - 1*p), 0, lty = 2)

### Compute Weights
d <- c(50,100,200,400)
vard <- matrix(nrow=1,ncol=4)
```

```
for (i in 1:4){
  vard[,i]=var(rstudent(inter)[data$dist==d[i]])
}
vard=data.frame(vard)
colnames(vard)=d
w <- array(dim=n)
for (j in 1:n){
  distj = data[j,]$dist
  w[j] = vard[[as.character(distj)]]
}

### Reweighter Model
inter.reweight <- lm(I(time**lambda)~1+dist*(stroke+sex+course),
                     data=data,weights=1/w)

# large cook's distance
iw <- cooks.distance(inter.reweight) > (8/(n - 2*p))

### New Diagnostics
par(mfrow=c(2,2))
qqnorm(rstudent(inter.reweight), pch=1+15*iw, col=1+iw)
qqline(rstudent(inter.reweight))

plot(fitted.values(inter.reweight), rstudent(inter.reweight),
     pch = 1 + 15*iw, col = 1 + iw)

plot(hatvalues(inter.reweight), ylim=c(0,0.1),pch=1+15*iw, col=1+iw)
abline(2*p/n, 0, lty = 2)

plot(cooks.distance(inter.reweight), pch = 1+15*iw, col = 1+iw)
abline(8/(n - 1*p), 0, lty = 2)

### Check if any data point has large leverage (high influence)
any(hatvalues(inter.reweight)[which(iw)]>2*p/n)

### Fitted Values and True Values
pieces3 <- split(data,data[,c('sex','course','stroke')])
par(mfrow=c(1,3))

df1 <- pieces3[['F.Long.Backstroke']]
x1 <- seq(80,220,by=2)
data1 <- data.frame(dist=x1,sex='F',course='Long',stroke='Backstroke')
pred1 <- (predict.lm(inter.reweight,newdata=data1))**(1/lambda)
plot(x1,pred1,main='F.Long.Backstroke',xlab='Distance',ylab='Time')
points(df1$dist,df1$time,col=2,pch=16)

df2 <- pieces3[['M.Long.Backstroke']]
x2 <- seq(80,220,by=2)
data2 <- data.frame(dist=x1,sex='M',course='Long',stroke='Backstroke')
```

```
pred2 <- (predict.lm(inter.reweight,newdata=data2))**(1/lambda)
plot(x2,pred2,main='M.Long.Backstroke',xlab='Distance',ylab='Time')
points(df2$dist,df2$time,col=2,pch=16)

df3 <- pieces3[['F.Short.Breaststroke']]
x3 <- seq(40,220,by=2)
data3 <- data.frame(dist=x3,sex='F',course='Short',stroke='Breaststroke')
pred3 <- (predict.lm(inter.reweight,newdata=data3))**(1/lambda)
plot(x3,pred3,main='F.Short.Breaststroke',xlab='Distance',ylab='Time')
points(df3$dist,df3$time,col=2,pch=16)


### Make Predictions on New Races Data
newdf <- data.frame(name=c('RaceA','RaceB','RaceC','RaceD'),
                    dist=c(400,50,100,100),
                    stroke=c('Freestyle','Backstroke','Butterfly','Medley'),
                    sex='F',
                    course='Long')
pred.intervals <- (predict.lm(inter.reweight,newdata=newdf,
                              interval='prediction'))**(1/lambda)
```