The Maximum Agreement Prediction via the Concordance Correlation Coefficient: Applications

Taeho Kim*, George Luta†, Matteo Bottai‡, Pierre Chausse§, Gheorghe Doros¶, Edsel A. Pena‖

Abstract

This vignette reproduces the applications included in Kim et al. (2023). For more details on how to use the package, see the other vignette.

1 Eye Data Set

In opthalmology, the central subfield macular thickness (CSMT) mea- surements can be obtained by optical coherence tomography (OCT). Abedi et al. (2011) focused on two types of OCT: time-domain Stratus OCT, the most widely used model prior to 2006; and spectral-domain Cirrus OCT, a more advanced model. As Cirrus OCT replaces Stratus OCT as a marker, the agreement between the measurements from two methods is of interest to researchers in the field. For this purpose, Abedi et al. (2011) provided a comparison between the two approaches and obtained a CCC-based conversion function from the Cirrus OCT to the Stratus OCT.

In the data set, both OCTs were measured from 46 subjects, i.e., 92 eyes, but only 61% of these observations were selected based on the reliability of the OCTs (signal strength ≥ 6 for both approaches). This subset of the original dataset is included in te package:

```
library(malp)
data(eye)
```

We can reproduce Figure 12 by first transforming the data:

```
Cirrus2 <- eye$Cirrus - 60
Cirrus3 <- 0.76*eye$Cirrus - 0.51
```

We can verify that we obtain the same concordance correlation coefficients as in the paper:

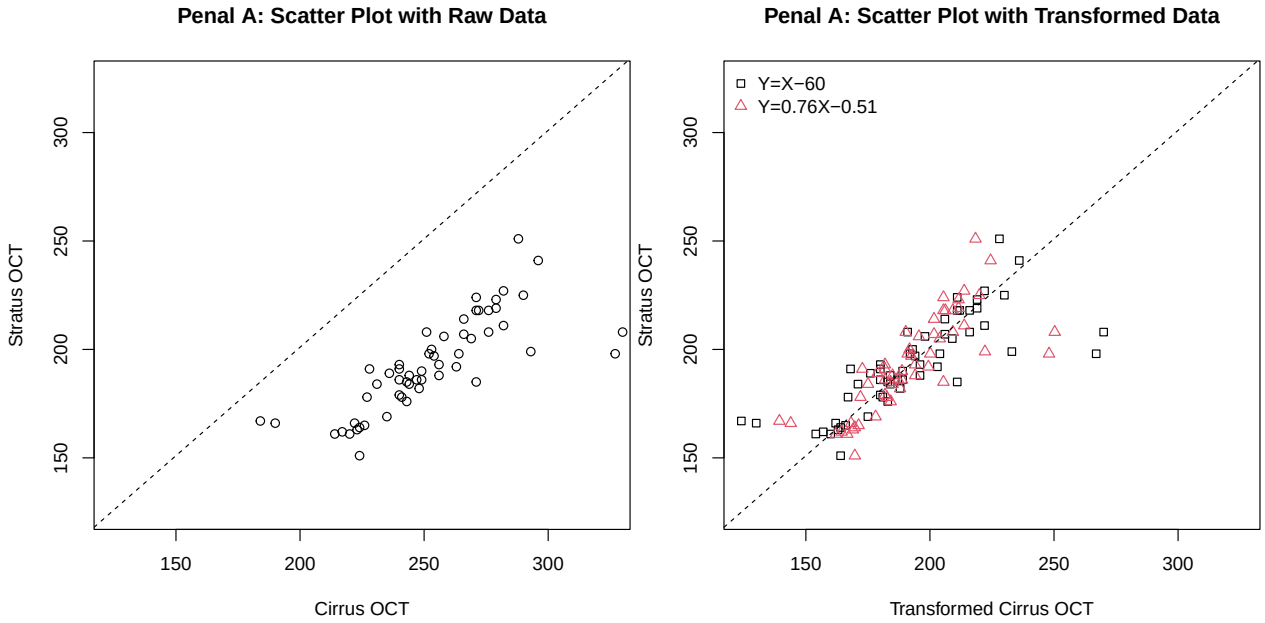
```
rho <- cor(eye$Stratus, eye$Cirrus)
rhoc <- c(ccc(eye$Stratus, eye$Cirrus),
          ccc(eye$Stratus, Cirrus2),
          ccc(eye$Stratus, Cirrus3))
res <- cbind(PCC=rho, CCC=rhoc)
rownames(res) <- c("Raw", "Transform 1" , "Transform 2")
knitr::kable(res)
```

	PCC	CCC
Raw	0.7834044	0.1998669
Transform 1	0.7834044	0.7561594
Transform 2	0.7834044	0.7813745

*Lehigh University, tak422@lehigh.edu
†Georgetown University, George.Luta@georgetown.edu
‡Karolinska Institutet, matteo.bottai@ki.se
§University of Waterloo, pchausse@uwaterloo.ca
¶Boston University,, doros@bu.edu
‖University of South Carolina, pena@stat.sc.edu

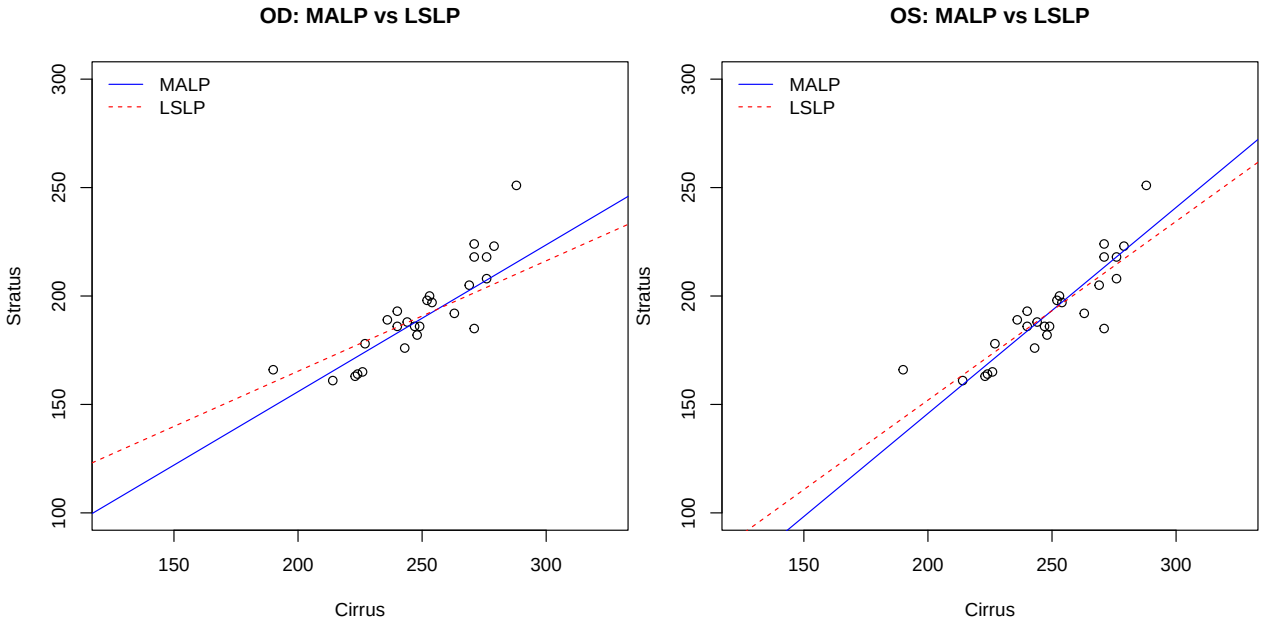
We can then reproduce Figure 12:

```
plot(Stratus~Cirrus, eye, xlab="Cirrus OCT", ylab="Stratus OCT",
     main="Penal A: Scatter Plot with Raw Data",
     xlim=c(125,325), ylim=c(125,325))
abline(1,1, lty=2)
plot(Stratus~Cirrus2, eye, xlab="Transformed Cirrus OCT",
     ylab="Stratus OCT", pch=22, xlim=c(125,325), ylim=c(125,325),
     main="Penal A: Scatter Plot with Transformed Data")
points(Cirrus3, eye$Stratus,pch=24, col=2)
abline(1,1, lty=2)
legend("topleft", c("Y=X-60","Y=0.76X-0.51"), col=1:2, pch=c(22,24),
       bty='n')
```



To produce Figure 13, we use the package to obtain the fitted lines for both MALP and LSLP:

```
eyeOD <- subset(eye, Eye=="OD")
eyeOS <- subset(eye, Eye=="OS")
fitOD <- malp(Stratus~Cirrus, eyeOD)
plot(Stratus~Cirrus, eyeOS, xlab="Cirrus", ylab="Stratus",
     main="OD: MALP vs LSLP", xlim=c(125,325), ylim=c(100,300))
abline(fitOD, lty=1, col="blue")
abline(fitOD$lm, lty=2, col="red")
legend("topleft", c("MALP","LSLP"), col=c("blue","red"), lty=1:2,bty='n')
fitOS <- malp(Stratus~Cirrus, eyeOS)
plot(Stratus~Cirrus, eyeOS, xlab="Cirrus", ylab="Stratus",
     main="OS: MALP vs LSLP", xlim=c(125,325), ylim=c(100,300))
abline(fitOS, lty=1, col="blue")
abline(fitOS$lm, lty=2, col="red")
legend("topleft", c("MALP","LSLP"), col=c("blue","red"), lty=1:2,bty='n')
```



We reproduce here the method to generate Table 3. The first function is to compute alll summary statistics needed (PCC, CCC and MSE):

```
## Realistic
getStat <- function(obj, newD)
{
    pr <- predict(obj, newdata=newD,includeLS=TRUE)
    malp <- c(cor(newD$Stratus, pr$MALP),
              ccc(newD$Stratus, pr$MALP),
              mean((newD$Stratus-pr$MALP)^2))
    lslp <- c(cor(newD$Stratus, pr$LSLP),
              ccc(newD$Stratus, pr$LSLP),
              mean((newD$Stratus-pr$LSLP)^2))
    c(malp, lslp)
}
```

The next function resample the data with replacement, splits the new sample in two, use the first half to estimate the model and the second half to compute the different statistics.

```
geti <- function(dat)
{
    n <- nrow(dat)
    dat <- dat[sample(n, replace=TRUE),]
    dat1 <- dat[1:floor(n/2),]
    dat2 <- dat[-(1:floor(n/2)),]
    fit <- malp(Stratus~Cirrus, dat1)
    getStat(fit, dat2)
}
```

The following is the bootstrap simulation for a given seed. It only runs if the simulation file “eyeSim.rda” is not on the current directory.

```
if (any(list.files("./") == "eyeSim.rda"))
{
    load("eyeSim.rda")
} else {
```

```
    set.seed(12052023)
    OS <- sapply(1:2000, function(i) geti(eyeOS))
    OD <- sapply(1:2000, function(i) geti(eyeOD))
    OS <- matrix(rowMeans(OS), ncol=2)
    OD <- matrix(rowMeans(OD), ncol=2)
    save(OD,OS,file="eyeSim.rda")
}
```

Finally, the following computes the results for what we refer to the "Illustrative" case, and combine all results in one table.

```
OSi <- matrix(getStat(fitOS, eyeOS), ncol=2)
ODi <- matrix(getStat(fitOD, eyeOD), ncol=2)
res <- cbind(OSi, ODi, OS, OD)
colnames(res) <- rep(c("MALP", "LSLP"),4)
rownames(res) <- c("PCC","CCC","MSE")
```

	Illustrative Case				Realistic Case			
	OS		OD		OS		OD	
	MALP	LSLP	MALP	LSLP	MALP	LSLP	MALP	LSLP
PCC	0.868	0.868	0.752	0.752	0.878	0.878	0.777	0.777
CCC	0.868	0.859	0.752	0.722	0.824	0.810	0.712	0.667
MSE	122.735	114.636	228.977	200.573	155.258	146.069	278.046	253.117

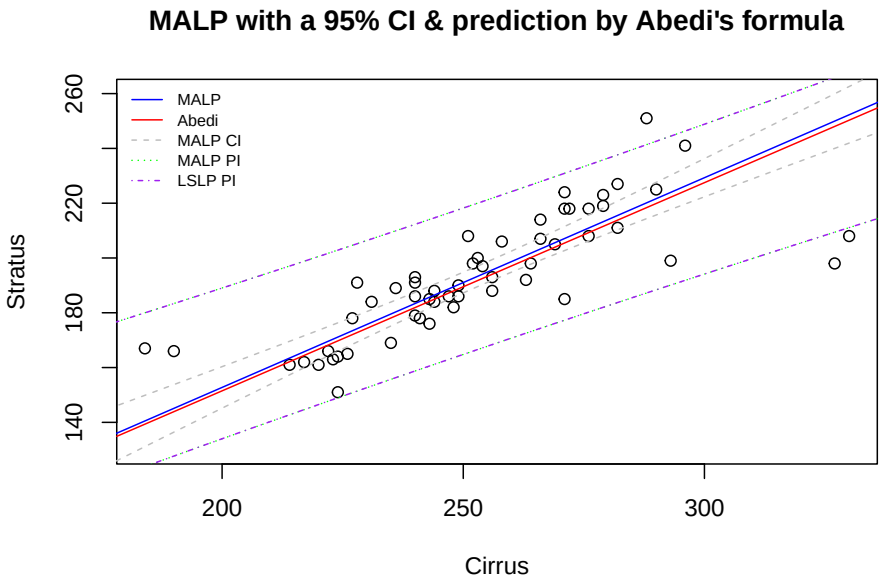
For Figure 14, we first estimate the model with all observations. We see that the coefficients are as shown in the paper.

```
fit <- malp(Stratus~Cirrus, eye)
coef(fit)
```

```
## (Intercept)      Cirrus
##  -0.3410832   0.7654732
```

To plot the confidence interval, we use the `predict` function and a grid of new Cirrus measures. In Figure 14, all standard errors are based on the asymptotic properties under normality. `matplot` is used to directly plot all lines from the `predict` output. For the prediction intervals, we only plot the lower and upper bounds, which correspond to the second and third row.

```
cir <- seq(120, 380, length.out=40)
pr <- predict(fit, newdata=data.frame(Cirrus=cir), interval="confidence",
              includeLS=TRUE)
pr2 <- predict(fit, newdata=data.frame(Cirrus=cir), interval="prediction",
              includeLS=TRUE)
plot(Stratus~Cirrus, eye, ylim=c(130,260),
     main="MALP with a 95% CI & prediction by Abedi's formula")
matplot(cir, pr$MALP, type='l',lty=c(1,2,2), col=c("blue", "gray", "gray"),
        add=TRUE)
matplot(cir, pr2$MALP[,-1], type='l',lty=3, col="green", add=TRUE)
matplot(cir, pr2$LSLP[,-1], type='l',lty=4, col="purple",add=TRUE)
abline(-.51, .76, col="red")
legend("topleft", c("MALP","Abedi","MALP CI", "MALP PI", "LSLP PI"),
       col=c("blue","red","gray","green","purple"), lty=c(1,1,2,3,4),
       bty='n', cex=.7)
```



2 The BodyFat Data set

The percentage of bodyfat is an important bodily characteristic that serves as a marker for the health status of an individual. Being able to infer it from easily-measured or determined bodily characteristics, such as age, weight, height, circumference measurements, or skin-fold measurements is therefore of interest, e.g., Behnke and Wilmore (1974), Katch and McArdle (1977). For example, one could obtain the percent bodyfat from body density via underwater weighting based on Siri's equation $100 \times \text{Bodyfat} = 495/\text{BodyDensity} - 450$ as in Siri (1956). Including body density (BD) and percent bodyfat (PBF), the data set of interest contains 13 additional variables such as age (years), weight (WGT, in pounds), height (HGT, in inches); several body circumference measurements (in cm): neck (NCK), chest (CST), abdomen (ABD), hip, thigh (TGH), knee (KN), ankle (ANK), biceps (BCP), forearm (FA), and wrist (WRT) for 252 men. The data set was originally in Penrose, Nelson, and Fisher (1985) and became available to the public by courtesy of Dr. A. Garth Fisher.

The dataset is included in the package and can be loaded as follows:

```
data(bodyFat)
```

2.1 Table 4

The table only returns the PCC between the different variables included in the dataset.

```
library(xtable)
corDat <- cor(bodyFat)
print(xtable(corDat, digits=2), floating=FALSE,size='small',
      hline=c(-1,-1,0,nrow(corDat),nrow(corDat)), comment=FALSE)
```

	BD	PBF	Age	WGT	HGT	NCK	CST	ABD	Hip	TGH	KN	ANK	BCP	FA	WRT
BD	1.00	-0.99	-0.28	-0.59	0.10	-0.47	-0.68	-0.80	-0.61	-0.55	-0.50	-0.26	-0.49	-0.35	-0.33
PBF	-0.99	1.00	0.29	0.61	-0.09	0.49	0.70	0.81	0.63	0.56	0.51	0.27	0.49	0.36	0.35
Age	-0.28	0.29	1.00	-0.01	-0.17	0.11	0.18	0.23	-0.05	-0.20	0.02	-0.11	-0.04	-0.09	0.21
WGT	-0.59	0.61	-0.01	1.00	0.31	0.83	0.89	0.89	0.94	0.87	0.85	0.61	0.80	0.63	0.73
HGT	0.10	-0.09	-0.17	0.31	1.00	0.25	0.13	0.09	0.17	0.15	0.29	0.26	0.21	0.23	0.32
NCK	-0.47	0.49	0.11	0.83	0.25	1.00	0.78	0.75	0.73	0.70	0.67	0.48	0.73	0.62	0.74
CST	-0.68	0.70	0.18	0.89	0.13	0.78	1.00	0.92	0.83	0.73	0.72	0.48	0.73	0.58	0.66
ABD	-0.80	0.81	0.23	0.89	0.09	0.75	0.92	1.00	0.87	0.77	0.74	0.45	0.68	0.50	0.62
Hip	-0.61	0.63	-0.05	0.94	0.17	0.73	0.83	0.87	1.00	0.90	0.82	0.56	0.74	0.55	0.63
TGH	-0.55	0.56	-0.20	0.87	0.15	0.70	0.73	0.77	0.90	1.00	0.80	0.54	0.76	0.57	0.56
KN	-0.50	0.51	0.02	0.85	0.29	0.67	0.72	0.74	0.82	0.80	1.00	0.61	0.68	0.56	0.66
ANK	-0.26	0.27	-0.11	0.61	0.26	0.48	0.48	0.45	0.56	0.54	0.61	1.00	0.48	0.42	0.57
BCP	-0.49	0.49	-0.04	0.80	0.21	0.73	0.73	0.68	0.74	0.76	0.68	0.48	1.00	0.68	0.63
FA	-0.35	0.36	-0.09	0.63	0.23	0.62	0.58	0.50	0.55	0.57	0.56	0.42	0.68	1.00	0.59
WRT	-0.33	0.35	0.21	0.73	0.32	0.74	0.66	0.62	0.63	0.56	0.66	0.57	0.63	0.59	1.00

2.2 Table 6

First, we estimate the five models represented by Table 5.

```
fit1 <- malp(PBF~ABD, bodyFat)
fit2 <- malp(PBF~ABD+WGT, bodyFat)
fit3 <- malp(PBF~ABD+WGT+FA+WRT, bodyFat)
fit4 <- malp(PBF~ABD+WGT+FA+WRT+Age+TGH, bodyFat)
fit5 <- malp(PBF~ABD+WGT+FA+WRT+Age+TGH+NCK+Hip, bodyFat)
```

We then organize the result in a table. The `summary` method with `se=FALSE` is used to compute the CCC, PCC and MSE.

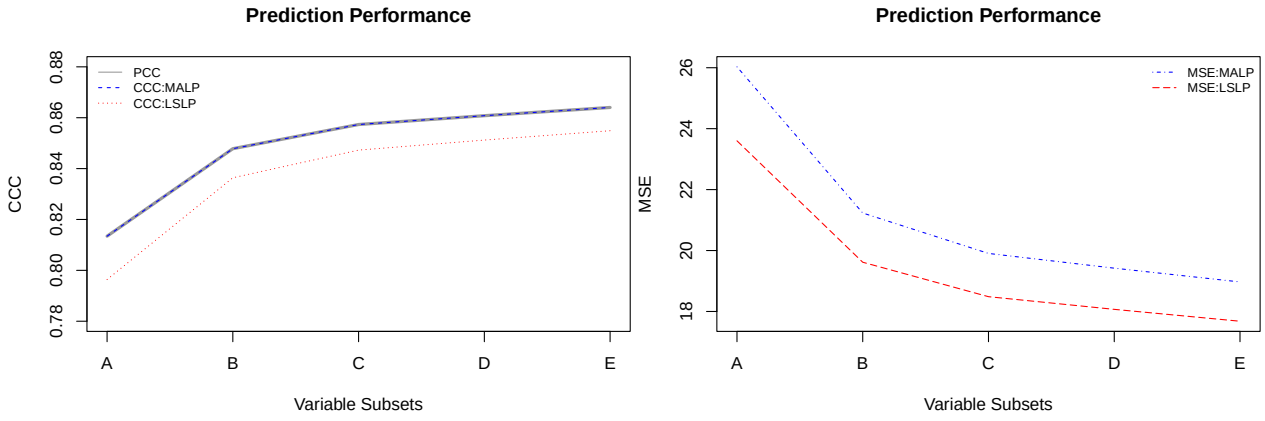
```
res <- lapply(1:5, function(i) {
    f <- get(paste("fit",i,sep=""))
    ans <- matrix(NA, 9+3, 2)
    s <- summary(f, se=FALSE)
    ans[1:length(coef(f)),] <- cbind(coef(f), coef(f$lm))
    ans[10:12,] <- cbind(s$fitMALP, s$fitLSLP)
    ans
    })
res <- do.call("cbind", res)
rownames(res) <- c(names(coef(fit5)), names(summary(fit5,se=FALSE)$fitMALP))
```

	A		B		C		D		E	
	MALP	LSLP	MALP	LSLP	MALP	LSLP	MALP	LSLP	MALP	LSLP
(Intercept)	-52.682	-39.280	-57.638	-45.952	-43.841	-34.854	-47.616	-38.322	-29.235	-22.656
ABD	0.776	0.631	1.167	0.990	1.161	0.996	1.059	0.912	1.093	0.945
WGT			-0.175	-0.148	-0.158	-0.136	-0.159	-0.136	-0.104	-0.090
FA					0.552	0.473	0.568	0.489	0.597	0.516
WRT					-1.756	-1.506	-2.067	-1.779	-1.778	-1.537
Age							0.073	0.063	0.076	0.066
TGH							0.256	0.220	0.350	0.302
NCK									-0.540	-0.467
Hip									-0.226	-0.195
PCC	0.813	0.813	0.848	0.848	0.857	0.857	0.861	0.861	0.864	0.864
CCC	0.813	0.796	0.848	0.836	0.857	0.847	0.861	0.851	0.864	0.855
MSE	26.029	23.601	21.232	19.616	19.905	18.485	19.421	18.070	18.969	17.680

2.3 Figure 15

We can use the info from the above table to produce the graph. But, instead of having the MSE and CCC on the same graph, which requires two different y-axes, we plot them separately.

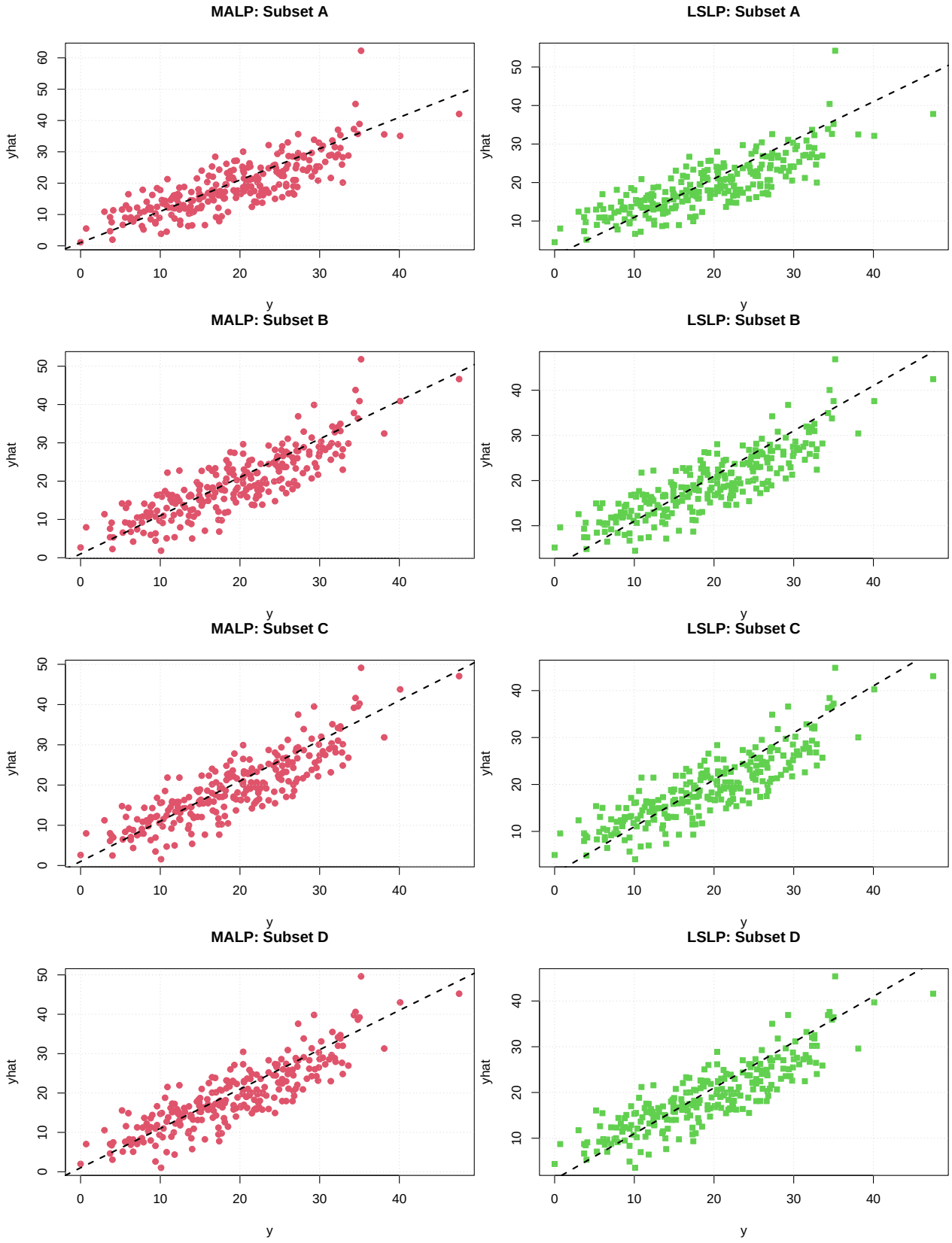
```
id <- seq(1,9,by=2)
plot(1:5, res["PCC",id], xaxt='n', main="Prediction Performance",
     xlab="Variable Subsets", ylab="CCC", ylim=c(.78,.88),
     type='l', lwd=3, col=gray(.2,.5))
axis(1, at=1:5, labels=LETTERS[1:5])
lines(1:5, res["CCC",id], col="blue", lty=2)
lines(1:5, res["CCC",id+1], col="red", lty=3)
legend("topleft", c("PCC","CCC:MALP","CCC:LSLP"),
       lty=c(1:3), col=c(gray(.2,.5),"blue","red"), cex=.75,
       bty='n')
plot(1:5, res["MSE",id], xaxt='n', main="Prediction Performance",
     xlab="Variable Subsets", ylab="MSE", ylim=range(res["MSE",]),
     type='l', lty=4, col="blue")
axis(1, at=1:5, labels=LETTERS[1:5])
lines(1:5, res["MSE",id+1], col="red", lty=5)
legend("topright", c("MSE:MALP","MSE:LSLP"),
       lty=c(4:5), col=c("blue","red"), cex=.75, bty='n')
```

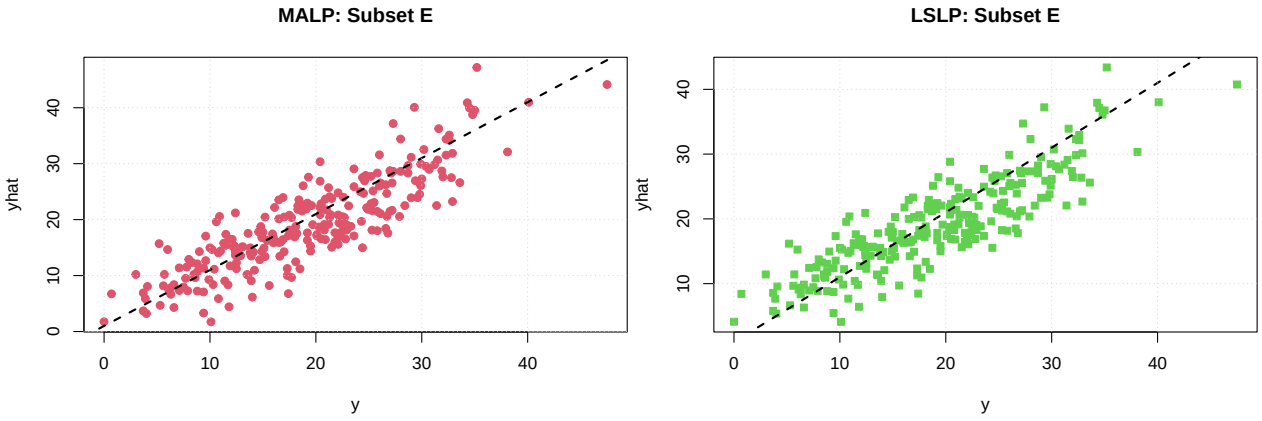


2.4 Figure 16

We can use the `plot` method form the package to produce all scatterplots.

```
plot(fit1, which="MALP", main="MALP: Subset A")
plot(fit1, which="LSLP", main="LSLP: Subset A")
plot(fit2, which="MALP", main="MALP: Subset B")
plot(fit2, which="LSLP", main="LSLP: Subset B")
plot(fit3, which="MALP", main="MALP: Subset C")
plot(fit3, which="LSLP", main="LSLP: Subset C")
plot(fit4, which="MALP", main="MALP: Subset D")
plot(fit4, which="LSLP", main="LSLP: Subset D")
plot(fit5, which="MALP", main="MALP: Subset E")
plot(fit5, which="LSLP", main="LSLP: Subset E")
```


MALP: Subset A
LSLP: Subset A
MALP: Subset B
LSLP: Subset B
MALP: Subset C
LSLP: Subset C
MALP: Subset D
LSLP: Subset D
yhat
y



References

Abedi, Gelareh, Payal Patal, Gheorghe Doros, and Manju L Subramanian. 2011. "Transitioning from Stratus OCT to Cirrus OCT: A Comparison and a Proposed Equation to Convert Central Subfield Macular Thickness Measurements in Healthy Subjects." *Graefe's Archive for Clinical and Experimental Ophthalmology* 249 (9): 1353–7.

Behnke, Albert Richard, and Jack H Wilmore. 1974. *Evaluation and Regulation of Body Build and Composition*. Englewood Cliffs, NJ: Prentice Hall.

Katch, Frank I, and William D McArdle. 1977. *Nutrition, Weight Control, and Exercise*. Boston, MA: Houghton Mifflin Company.

Kim, Taeho, George Luta, Matteo Bottai, Pierre Chausse, Gheorghe Doros, and Edsel A. Pena. 2023. "Maximum Agreement Linear Prediction via the Concordance Correlation Coefficient." http://arxiv.org/abs/2304.04221.

Penrose, Keith W, AG Nelson, and AG Fisher. 1985. "Generalized Body Composition Prediction Equation for Men Using Simple Measurement Techniques." *Medicine & Science in Sports & Exercise* 17 (2): 189. http://lib.stat.cmu.edu/datasets/bodyfat.

Siri, William E. 1956. "The Gross Composition of the Body." In *Advances in Biological and Medical Physics*, 4:239–80. Elsevier.