

Model Fit Criteria

Applied to GWAS data (one genotype) from lab

Data

```
genos = as.matrix(read.table("./genos.txt"))
g1 = genos[,1]
table(g1)
phenos = as.matrix(read.table("./phenos.txt"))
hist(phenos)
summary(phenos)
```

Fitting Models

Fit linear model with `lm()`

```
mod = lm(phenos~g1)
summary(mod)
names(mod)
length(phenos)
mod$df.residual
```

Fit a linear model with `glm()`

```
glm(phenos~g1,family="gaussian")
mod
```

More on the `lm()` function

```
? lm
lm(phenos~0+g1)
lm(phenos~g1-1)
lm(phenos~1)
```

Compare to coding `g1` as a factor rather than integer

```
summary(lm(phenos~as.factor(g1)))

#visualization
#observed outcome and covariate
boxplot(phenos~g1,xlab="genotype",ylab="phenotype")

#predicted outcome and covariate
plot(g1,mod$fitted.values,xlab="genotype",ylab="predicted phenotype")

#observed vs. predicted
plot(phenos,mod$fitted.values,xlab="observed phenotype",ylab="predicted phenotype",pch=as.character(g1))

#residuals vs. covariate
plot(g1,mod$residuals,xlab="genotype",ylab="residuals")

#qq plot residuals vs. normal quantiles
qqnorm(mod$residuals,ylab="Residuals")
```

Quantitative criteria

r-squared, adjusted r-squared, residual standard error

```
summary(mod)
```

Compare to a model with two genotypes and interaction

```
g2 = genos[,2]
table(g2)
mod2 = lm(phenos~g1+g2+g1:g2)
summary(mod2)
summary(lm(phenos~g1*g2))
```

AIC and forward variable selection

```
#AIC = -2 log likelihood + 2*rank of model
mod0 = lm(phenos~1)
step(mod0,scope=list(lower=mod0,upper=mod2),direction=c("forward"))
#AIC and backward variable selection
step(mod2,scope=list(lower=mod0,upper=mod2),direction=c("backward"))
```

Cross validation prediction error

```
#install.packages("boot")
library(boot)
?cv.glm
```

```
df = data.frame(phenos=as.vector(phenos),g1=g1,g2=g2)
cv1 = cv.glm(df,glm(phenos~g1,family="gaussian"),K=5)
cv1$delta
cv.glm(df,glm(phenos~g1,family="gaussian"))$delta
cv2 = cv.glm(df,glm(phenos~g1+g2,family="gaussian"),K=5)
cv2$delta
cv3 = cv.glm(df,glm(phenos~g1*g2,family="gaussian"),K=5)
cv3$delta
#the interaction may be overfitting
```

Penalized regression

On first 20 genotypes (for computational efficiency, can do all)

```
library(glmnet)
```

Ridge regression: L1 penalty, small coefficients

```
lambda1=cv.glmnet(x=genos[,1:20],y=phenos,family="gaussian",alpha=0)$lambda.min
lambda1
mod3 = glmnet(x=genos[,1:20],y=phenos,family="gaussian",alpha=0,lambda=lambda1)
mod3$beta
```

Lasso regression: L2 penalty, zero coefficients

```
lambda2 = cv.glmnet(x=genos[,1:20],y=phenos,family="gaussian",alpha=1)$lambda.min
lambda2
mod4 = glmnet(x=genos[,1:20],y=phenos,family="gaussian",alpha=1,lambda=lambda2)
mod4$beta
```

Elastic net regression: combined penalty

```
lambda3 = cv.glmnet(x=genos[,1:20],y=phenos,family="gaussian",alpha=0.5)$lambda.min
lambda3
mod5 = glmnet(x=genos[,1:20],y=phenos,family="gaussian",alpha=0.5,lambda=lambda3)
cbind(mod3$beta,mod4$beta,mod5$beta)
```