# 1.客运量与机车台数的线性回归检验

客运量y=c+b机车台数

> lm.sol<-lm(PV~locomotive,data=passanger)

> lm.sol

Call:

lm(formula = PV ~ locomotive, data = passanger)

Coefficients:

(Intercept) locomotive

-263664.62 22.33

> summary(lm.sol)

Call:

lm(formula = PV ~ locomotive, data = passanger)

Residuals:

Min 1Q Median 3Q Max

-12850 -9147 -4119 6262 34797

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -2.637e+05 3.922e+04 -6.722 2.13e-05 \*\*\*

locomotive 2.233e+01 2.113e+00 10.567 1.97e-07 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 14050 on 12 degrees of freedom

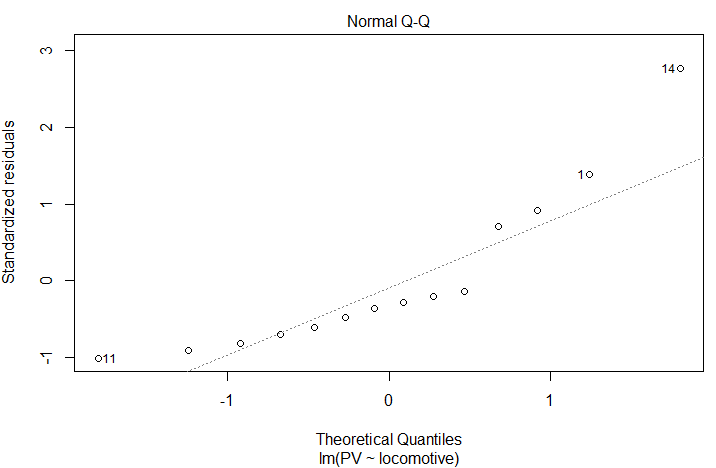
Multiple R-squared: 0.903, Adjusted R-squared: 0.8949

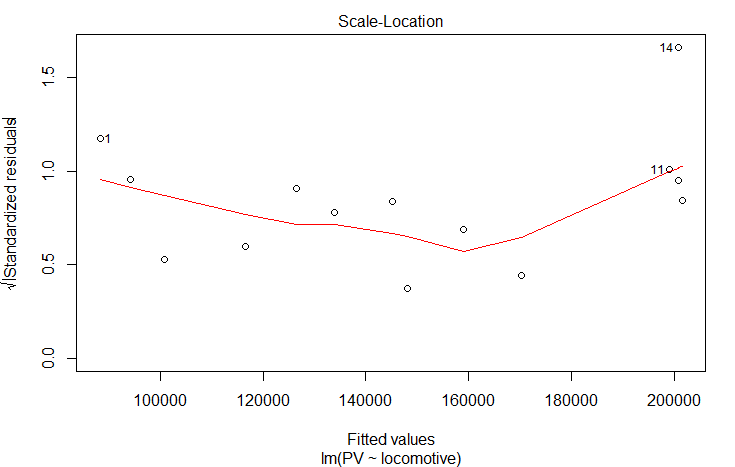
F-statistic: 111.7 on 1 and 12 DF, p-value: 1.967e-07

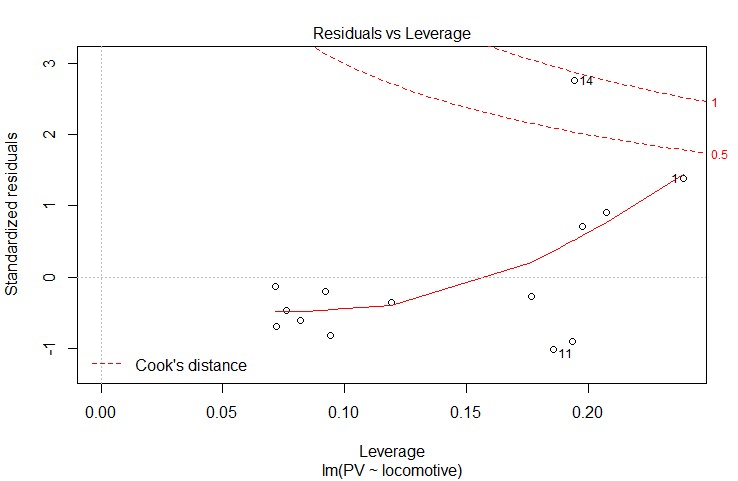
1. 显著性检验：R2为0.903，F检验p值小于0.05，通过显著性检验。模型中机车台数的系数3颗星，通过检验。

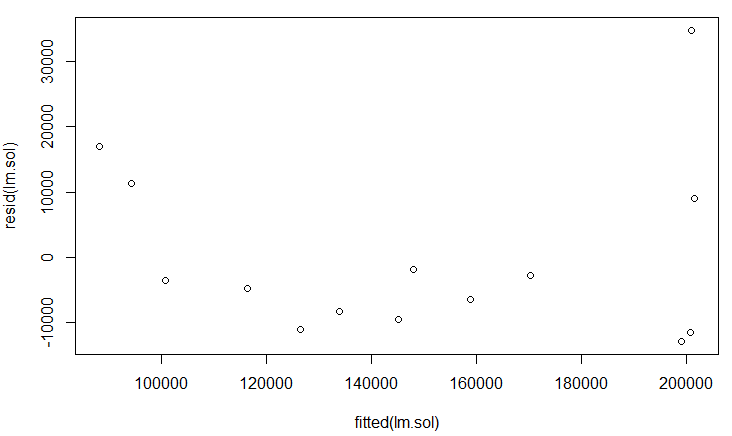
2. 共线性检验：因为只有一个变量，x1只有1个，故不存在共线性检验这个步骤；VIF方差膨胀因子是衡量共线性的指标

3. GQ异方差检验-残差方差是相等的，方法有PP图 QQ图等；







残差与回归值的散点图

# 2.客运量与机车台数的随机森林检验

passanger<-read.csv("locomotive-PV.csv",head=T)

randomForest(PV~locomotive,data=passanger,ntree=1,importance=TRUE,proximity=T)

跑出来结果是：

Call:

randomForest(formula = PV ~ locomotive, data = passanger, ntree = 1, importance = TRUE, proximity = T)

Type of random forest: regression

Number of trees: 1

No. of variables tried at each split: 1

Mean of squared residuals: 516826577

% Var explained: 70.34

方差解释为70.34%，不是很好哇，

检验：代码及结果

> names(passanger)

[1] "tm" "locomotive" "PV"

> dim(passanger)

[1] 14 3

> n=dim(passanger)[i]

> m=sample(1:n,ceiling(n/2))

> n<-100

> NMSE<-rep(0,n)

> NEMSE0<-NMSE

> set.seed(100)

for(i in 1:n){A=randomForest(PV~locomotive,data=passanger[-m,],importance=T,proximity=T,ntree=i);

+ y0=predict(A,passanger[-m,]);

+ y1=predict(A,passanger[m,]);

+ NMSE0[i]<-mean((passanger$PV[-m]-y0)^2)/mean((passanger$PV[-m]-mean(passanger$PV[-m]))^2);

+ NMSE[i]=mean((passanger$PV-y1)^2)/mean((passanger$PV-mean(passanger$PV[m]))^2)}

>MNMSE0<-mean(NMSE0)

> MNMSE<-MEAN(NMSE)

> MNMSE0

[1] 0.1114241

> MNMSE

[1] 2.060804

两者差异很大，通不过

# 3.客运量与机车台数的支持向量机检验

svmReg<-function(dbase,model){

#Support vector machine, SVM

#require(rminer)

require(e1071)

n<-dim(dbase)[1]

m=sample(1:n,ceiling(n/2))

n<-500

NMSE<-rep(0,n)

NMSE0<-NMSE

set.seed(1445)

for(i in 1:n){

# M<-fit(f~x1+x2+x3+x4,data=dbase,model="svm")

M<-svm(model,data=dbase[-m,]) #??e1071??????

y0<-predict(M,dbase[-m,]);

y1<-predict(M,dbase[m,]);

NMSE0[i]<-mean((dbase$PV[-m]-y0)^2)/mean((dbase$PV[-m]-mean(dbase$PV[-m]))^2);

NMSE[i]<-mean((dbase$PV[m]-y1)^2)/mean((dbase$PV[m]-mean(dbase$PV[m]))^2)

}

print(M)

MNMSE0<-mean(NMSE0)

MNMSE<-mean(NMSE)

cat("SVM训练集NMSE0=",MNMSE0,"\n")

cat("测试集NMSE=",MNMSE,"\n")

par(mfrow=c(1,1))

plot(1:n,NMSE,type="l",ylim=c(min(NMSE,NMSE0),max(NMSE,NMSE0)),ylab="NMSE",main="Support Vector Machine:NMSE",lty=2)

lines(1:n,NMSE0)

legend("topright",c("Training Set","Testing Set"),lty=1:2)

n<-dim(dbase)[1]

f.fit<-rep(0,n)

for(i in 1:n){

f.fit[i]<-predict(M,dbase[i,])

}

plot(1:n,f.fit,type="p",pch=1,

ylim=c(min(f.fit,dbase[,1]),max(f.fit,dbase[,1])),

ylab="Railway Freight",xlab="Date",main="支持向量机回归预测")

lines(1:n,dbase$f,pch=16)

results<-as.matrix(f.fit,ncol=1)

}

passanger<-read.csv("locomotive-PV.csv",head=T)

svmReg(passanger,locomotive~PV)

结果为

Call:

svm(formula = model, data = dbase[-m, ])

Parameters:

SVM-Type: eps-regression

SVM-Kernel: radial

cost: 1

gamma: 1

epsilon: 0.1

Number of Support Vectors: 5

SVM训练集NMSE0= 15.47662

测试集NMSE= 10.67733

差异挺大的。