1. Multiple y format (LR)

```
# read data ------------------------------------------------------------------

X_train <- read.csv("ModelData/X_train2.csv")[,-1]; Y_train <- read.csv("ModelData/Y_train2.csv")[,-1]
X_val <- read.csv("ModelData/X_val2.csv")[,-1]; Y_val <- read.csv("ModelData/Y_val2.csv")[,-1]
X_test <- read.csv("ModelData/X_test2.csv")[,-1]; Y_test <- read.csv("ModelData/Y_test2.csv")[,-1]

colnames(Y_train) <- c("T1","T2","T3","T4","T5","T6","T7")
Data_train <- cbind(Y_train,X_train); #head(Data_train); dim(Data_train)

colnames(Y_val) <- c("T1","T2","T3","T4","T5","T6","T7")
Data_val <- cbind(Y_val,X_val); #head(Data_val); dim(Data_val)

colnames(Y_test) <- c("T1","T2","T3","T4","T5","T6","T7")
Data_test <- cbind(Y_test,X_test); #head(Data_test); dim(Data_test)
```

1.1. Linear regression without interactions

```
# LR ------------------------------------------------------------------

ml <- lm(cbind(Data_train$Y1,Data_train$Y2,Data_train$Y3,Data_train$Y4,Data_train$Y5,Data_train$Y6,Data_train$Y7)
~ . , Data_train[,-c(1:7)])
#summary(ml)
#cat("RMSE:", mean(ml$residuals^2))
#cat("RMSE:", sqrt(mean(ml$residuals^2)))

y_pred <- predict(ml, Data_val)
y_real <- Y_val
y_pred <- rbind(y_pred[,1],y_pred[,2],y_pred[,3],y_pred[,4],y_pred[,5],y_pred[,6],y_pred[,7])
y_real <- rbind(y_real[,1],y_real[,2],y_real[,3],y_real[,4],y_real[,5],y_real[,6],y_real[,7])
cat("MSE:", mean(y_real-y_pred)^2)
```

```
## MSE: 0.1117158
```

```
cat("RMSE:", sqrt(mean(y_real-y_pred)^2))
```

```
## RMSE: 0.3342391
```

```
y_pred <- predict(ml, Data_test)
y_real <- Y_test
y_pred <- rbind(y_pred[,1],y_pred[,2],y_pred[,3],y_pred[,4],y_pred[,5],y_pred[,6],y_pred[,7])
y_real <- rbind(y_real[,1],y_real[,2],y_real[,3],y_real[,4],y_real[,5],y_real[,6],y_real[,7])
cat("MSE:", mean(y_real-y_pred)^2)
```

```
## MSE: 0.09493806
```

```
cat("RMSE:", sqrt(mean(y_real-y_pred)^2))
```

```
## RMSE: 0.3081202
```

2. One y format, "all var & station" interactions (LR, LASSO, Ridge, Decision Tree)

```
# read data ------------------------------------------------------------------
Train <- read.csv("ModelData/Train2_long.csv"); #head(Train); dim(Train)
Val <- read.csv("ModelData/Val2_long.csv"); #head(Val); dim(Val)
Test <- read.csv("ModelData/Test2_long.csv"); #head(Test); dim(Test)

# categorical var
for(i in 4:14){
  Train[,i] <- as.character(Train[,i])
  Val[,i] <- as.character(Val[,i])
  Test[,i] <- as.character(Test[,i])
}
median_test <- median(Test[,3]); iqr_test <- IQR(Test[,3])
Train <- Train[,-1]; Val <- Val[,-1]; Test <- Test[,-1]
#summary(Train); summary(Val); summary(Test)
```

2.1. Linear regression with interactions

```
lr_model <- lm(mrt_flow~ . * mrt_station, Train)
#lr_model <- lm(mrt_flow ~ . * mrt_station, rbind(Train,Val))
#summary(lr_model)
#cat("MSE:", mean(mlr$residuals^2))
#cat("RMSE:", sqrt(mean(mlr$residuals^2)))

y_pred <- predict(lr_model, Val)
```

```
## Warning in predict.lm(lr_model, Val): prediction from a rank-deficient fit may
## be misleading
```

```
y_real <- Val$mrt_flow
cat("MSE:", mean(y_real-y_pred)^2)
```

```
## MSE: 0.104414
```

```
cat("RMSE:", sqrt(mean(y_real-y_pred)^2))
```

```
## RMSE: 0.3231316
```

```
cat("Original RMSE:", sqrt(mean(( y_real+iqr_test+median_test) - (y_pred+iqr_test+median_test) )^2))
```

```
## Original RMSE: 725.4304
```

```
y_pred <- predict(lr_model, Test)
```

```
## Warning in predict.lm(lr_model, Test): prediction from a rank-deficient fit may
## be misleading
```

```
y_real <- Test$mrt_flow
cat("MSE:", mean(y_real-y_pred)^2)
```

```
## MSE: 0.06850735
```

```
cat("RMSE:", sqrt(mean(y_real-y_pred)^2))
```

```
## RMSE: 0.2617391
```

```
cat("Original RMSE:", sqrt(mean(( y_real+iqr_test+median_test) - (y_pred+iqr_test+median_test) )^2))
```

```
## Original RMSE: 587.6042
```

2.2. LASSO with interactions

```
f <- as.formula(mrt_flow ~ .* mrt_station) # using .*. for all interactions
y <- Train$mrt_flow
x <- model.matrix(f, Train)[,-1] # using model.matrix to take advantage of f
#y <- rbind(Train,Val)$mrt_flow
#x <- model.matrix(f, rbind(Train,Val))[,-1] # using model.matrix to take advantage of f

library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1-7
```

```
lasso_kfold <- cv.glmnet(x, y, alpha=1, nfolds=10)
lasso_best_lambda <- lasso_kfold$lambda.min
lasso_model <- glmnet(x, y, alpha=1, lambda=lasso_best_lambda)

x <- model.matrix(mrt_flow ~.*mrt_station, rbind(Train,Val))[,-1][-(1:nrow(Train)),]
y_pred <- predict(lasso_model, x)
y_real <- Val$mrt_flow
cat("MSE:", mean(y_real-y_pred)^2)
```

```
## MSE: 0.1019242
```

```
cat("RMSE:", sqrt(mean(y_real-y_pred)^2))
```

```
## RMSE: 0.3192557
```

```
cat("Original RMSE:", sqrt(mean(( y_real+iqr_test+median_test) - (y_pred+iqr_test+median_test) )^2))
```

```
## Original RMSE: 716.729
```

```
x <- model.matrix(mrt_flow ~.*mrt_station, rbind(Train,Test))[,-1][-(1:nrow(Train)),]
#x <- model.matrix(mrt_flow ~.*mrt_station, rbind(Train,Val,Test))[,-1][-(1:nrow(rbind(Train,Val))),]
y_pred <- predict(lasso_model, x)
y_real <- Test$mrt_flow
cat("MSE:", mean(y_real-y_pred)^2)
```

```
## MSE: 0.07014739
```

```
cat("RMSE:", sqrt(mean(y_real-y_pred)^2))
```

```
## RMSE: 0.2648535
```

```
cat("Original RMSE:", sqrt(mean(( y_real+iqr_test+median_test) - (y_pred+iqr_test+median_test) )^2))
```

```
## Original RMSE: 594.5962
```

2.3. Ridge with interactions

```
f <- as.formula(mrt_flow ~ .* mrt_station) # using .*. for all interactions
y <- Train$mrt_flow
x <- model.matrix(f, Train)[,-1] # using model.matrix to take advantage of f
#y <- rbind(Train,Val)$mrt_flow
#x <- model.matrix(f, rbind(Train,Val))[,-1] # using model.matrix to take advantage of f

library(glmnet)
ridge_kfold <- cv.glmnet(x, y, alpha=1, nfolds=10)
ridge_best_lambda <- ridge_kfold$lambda.min
ridge_model <- glmnet(x, y, alpha=0, lambda=ridge_best_lambda)

x <- model.matrix(mrt_flow ~.*mrt_station, rbind(Train,Val))[,-1][-(1:nrow(Train)),]
y_pred <- predict(ridge_model, x)
y_real <- Val$mrt_flow
cat("MSE:", mean(y_real-y_pred)^2)
```

```
## MSE: 0.1013714
```

```
cat("RMSE:", sqrt(mean(y_real-y_pred)^2))
```

```
## RMSE: 0.3183888
```

```
cat("Original RMSE:", sqrt(mean(( y_real+iqr_test+median_test) - (y_pred+iqr_test+median_test) )^2))
```

```
## Original RMSE: 714.7828
```

```
x <- model.matrix(mrt_flow ~.*mrt_station, rbind(Train,Test))[,-1][-(1:nrow(Train)),]
#x <- model.matrix(mrt_flow ~.*mrt_station, rbind(Train,Val,Test))[,-1][-(1:nrow(rbind(Train,Val))),]
y_pred <- predict(ridge_model, x)
y_real <- Test$mrt_flow
cat("MSE:", mean(y_real-y_pred)^2)
```

```
## MSE: 0.06688835
```

```
cat("RMSE:", sqrt(mean(y_real-y_pred)^2))
```

```
## RMSE: 0.2586278
```

```
cat("Original RMSE:", sqrt(mean(( y_real+iqr_test+median_test) - (y_pred+iqr_test+median_test) )^2))
```

```
## Original RMSE: 580.6195
```

2.4. GLMNET (combine LASSO and Ridge) with interactions

```
f <- as.formula(mrt_flow ~ .* mrt_station) # using .*. for all interactions
y <- Train$mrt_flow
x <- model.matrix(f, Train)[,-1] # using model.matrix to take advantage of f
#y <- rbind(Train,Val)$mrt_flow
#x <- model.matrix(f, rbind(Train,Val))[,-1] # using model.matrix to take advantage of f

library(glmnet)
ridge_kfold <- cv.glmnet(x, y, alpha=0.05, nfolds=10) # choose alpha by the performance in val
ridge_best_lambda <- ridge_kfold$lambda.min
ridge_model <- glmnet(x, y, alpha=0, lambda=ridge_best_lambda)

x <- model.matrix(mrt_flow ~.*mrt_station, rbind(Train,Val))[,-1][-(1:nrow(Train)),]
y_pred <- predict(ridge_model, x)
y_real <- Val$mrt_flow
cat("MSE:", mean(y_real-y_pred)^2)
```

```
## MSE: 0.1012653
```

```
cat("RMSE:", sqrt(mean(y_real-y_pred)^2))
```

```
## RMSE: 0.3182221
```

```
cat("Original RMSE:", sqrt(mean(( y_real+iqr_test+median_test) - (y_pred+iqr_test+median_test) )^2))
```

```
## Original RMSE: 714.4087
```

```
x <- model.matrix(mrt_flow ~.*mrt_station, rbind(Train,Test))[,-1][-(1:nrow(Train)),]
#x <- model.matrix(mrt_flow ~.*mrt_station, rbind(Train,Val,Test))[,-1][-(1:nrow(rbind(Train,Val))),]
y_pred <- predict(ridge_model, x)
y_real <- Test$mrt_flow
cat("MSE:", mean(y_real-y_pred)^2)
```

```
## MSE: 0.06683383
```

```
cat("RMSE:", sqrt(mean(y_real-y_pred)^2))
```

```
## RMSE: 0.2585224
```

```
cat("Original RMSE:", sqrt(mean(( y_real+iqr_test+median_test) - (y_pred+iqr_test+median_test) )^2))
```

```
## Original RMSE: 580.3828
```

2.5. Regression tree (decision tree)

```
library(rpart)
dt_model <- rpart(mrt_flow~., Train, cp=0.000005) # choose cp by the performance in val
#dt_model <- rpart(mrt_flow~.,rbind(Train,Val))
#summary(dt_model)
#printcp(dt_model)
#plotcp(dt_model)
#dt_model_pruned <- prune(dt_model, cp = 0.00001)

y_pred <- predict(dt_model, Val)
y_real <- Val$mrt_flow
cat("MSE:", mean(y_real-y_pred)^2)
```

```
## MSE: 0.053406
```

```
cat("RMSE:", sqrt(mean(y_real-y_pred)^2))
```

```
## RMSE: 0.2310974
```

```
cat("Original RMSE:", sqrt(mean(( y_real+iqr_test+median_test) - (y_pred+iqr_test+median_test) )^2))
```

```
## Original RMSE: 518.8136
```

```
y_pred <- predict(dt_model, Test)
#y_pred <- predict(dt_model_pruned, Test)
y_real <- Test$mrt_flow
cat("MSE:", mean(y_real-y_pred)^2)
```

```
## MSE: 0.02419636
```

```
cat("RMSE:", sqrt(mean(y_real-y_pred)^2))
```

```
## RMSE: 0.1555518
```

```
cat("Original RMSE:", sqrt(mean(( y_real+iqr_test+median_test) - (y_pred+iqr_test+median_test) )^2))
```

```
## Original RMSE: 349.2138
```