Chembl 21 Inhibitors Prediction

*Saw Simeon, Nuttapat Anuwongcharoen, Watshara Shoombuatong, Aijaz Ahmad Malik,
Virapong Prachayasittikul, Jarl E. S. Wikberg and Chanin Nantasenamat*

June 24, 2559 BE

Function for Reading external data set

```
read_file <- function(x) {
    library(data.table)
    data <- fread(x)
    IC50_nm <- data$IC50
    IC50 <- as.numeric(IC50_nm) * 10^-9
    pIC50 <- -log10(IC50)
    data <- as.data.frame(data)
    curated_data <- cbind(pIC50, data)
    return(curated_data)
}
```

Importing new AchE inhibitors from Chembl 21

```
AtomPairs2D_fingerPrintCount <- read_file("data_Chembl_21/2D_Atom_Paris_Count.csv")
AtomPairs2D_fingerPrinter <- read_file("data_Chembl_21/2D_Atom_Pairs.csv")
Substructure_fingerPrintCount <- read_file("data_Chembl_21/Substructure_Count.csv")
Substructure_fingerPrinter <- read_file("data_Chembl_21/Substructure.csv")
Extended_finterPrinter <- read_file("data_Chembl_21/CDK_Extended.csv")
FingerPrinter <- read_file("data_Chembl_21/CDK.csv")
Estate_FingerPrinter <- read_file("data_Chembl_21/E_State.csv")
GraphOnly_FingerPrinter <- read_file("data_Chembl_21/CDK_Graph_Only.csv")
KlekotaRoth_FingerprintCount <- read_file("data_Chembl_21/Klekota_Roth_Count.csv")
KlekotaRoth_FingerPrinter <- read_file("data_Chembl_21/Klekota_Roth.csv")
MACCS_FingerPrinter <- read_file("data_Chembl_21/MACCS.csv")
Pubchem_FingerPrinter <- read_file("data_Chembl_21/PubChem.csv")


external <- list(AtomPairs2D_fingerPrintCount = AtomPairs2D_fingerPrintCount,
    AtomPairs2D_fingerPrinter = AtomPairs2D_fingerPrinter, Substructure_fingerPrintCount = Substructure_
    Substructure_fingerPrinter = Substructure_fingerPrinter, Extended_finterPrinter = Extended_finterPr:
    FingerPrinter = FingerPrinter, Estate_FingerPrinter = Estate_FingerPrinter,
    GraphOnly_FingerPrinter = GraphOnly_FingerPrinter, KlekotaRoth_FingerprintCount = KlekotaRoth_Finge:
    KlekotaRoth_FingerPrinter = KlekotaRoth_FingerPrinter, MACCS_FingerPrinter = MACCS_FingerPrinter,
    Pubchem_FingerPrinter = Pubchem_FingerPrinter)
```

```
randomForest_testing <- function(train, test) {
    library(parallel)
    library(doSNOW)
    cl <- makeCluster(8)
    registerDoSNOW(cl)

    R2 <- function(y, equation, ...) {
```

```r
        1 - (sum((y - predict(equation))^2)/sum((y - mean(y))^2))
}
rm2 <- function(y, x, ...) {
    if ((R2(y, (lm(y ~ x)))) > R2(y, (lm(y ~ -1 + x)))) {
        return(R2(y, (lm(y ~ x))) * (1 - (sqrt(R2(y, (lm(y ~ x))) - R2(y,
            (lm(y ~ -1 + x)))))))
    } else {
        return(R2(y, (lm(y ~ x))))
    }
}
rm2.reverse <- function(y, x, ...) {
    return(R2(x, (lm(x ~ y))) * (1 - (sqrt(R2(x, (lm(x ~ y))) - R2(x, (lm(x ~
        -1 + y)))))))
}
average.rm2 <- function(y, x, ...) {
    if ((R2(y, (lm(y ~ x)))) > R2(y, (lm(y ~ -1 + x)))) {
        return(((R2(y, (lm(y ~ x))) * (1 - (sqrt(R2(y, (lm(y ~ x))) - R2(y,
            (lm(y ~ -1 + x)))))))) + R2(x, (lm(x ~ y))) * (1 - (sqrt(R2(x,
            (lm(x ~ y))) - R2(x, (lm(x ~ -1 + y)))))))/2)
    } else {
        return(((R2(y, (lm(y ~ x)))) + (R2(x, (lm(x ~ y))) * (1 - (sqrt(R2(x,
            (lm(x ~ y))) - R2(x, (lm(x ~ -1 + y)))))))/2)
    }
}
delta.rm2 <- function(y, x, ...) {
    if ((R2(y, (lm(y ~ x)))) > R2(y, (lm(y ~ -1 + x)))) {
        return(abs((R2(y, (lm(y ~ x))) * (1 - (sqrt(R2(y, (lm(y ~ x))) -
            R2(y, (lm(y ~ -1 + x)))))) - R2(x, (lm(x ~ y))) * (1 - (sqrt(R2(x,
            (lm(x ~ y))) - R2(x, (lm(x ~ -1 + y)))))))
    } else {
        return(abs((R2(y, (lm(y ~ x)))) - (R2(x, (lm(x ~ y))) * (1 - (sqrt(R2(x,
            (lm(x ~ y))) - R2(x, (lm(x ~ -1 + y)))))))
    }
}


results <- list(100)
results <- foreach(i = 1:100) %dopar% {

    x <- na.omit(train)
    para <- dplyr::sample_n(x, size = 2570, replace = TRUE)
    in_train_para <- sample(nrow(para), size = as.integer(nrow(para) * 0.8),
        replace = FALSE)
    Train <- para[in_train_para, ]
    unused <- para[-in_train_para, ]
    rm(unused)
    des <- Train[, 2:ncol(Train)]
    name <- names(des)
    Test <- test
    pIC50 <- Test$pIC50
    Test <- Test[, name]

    model_train <- ranger::ranger(pIC50 ~ ., data = Train, write.forest = TRUE,
```

```
            save.memory = TRUE)
        # actual <- train$Activity
        prediction <- predict(model_train, Test)
        prediction <- prediction$predictions
        value <- data.frame(obs = pIC50, pred = prediction)
        rm(Train)
        rm(Test)
        rm(para)
        rm(in_train_para)
        rm(prediction)
        labeling <- c("obs", "pred")
        colnames(value) <- labeling
        result <- caret::defaultSummary(value)
        result_rm2 <- rm2(value$obs, value$pred)
        names(result_rm2) <- "rm2"
        results_reverse <- rm2.reverse(value$obs, value$pred)
        names(results_reverse) <- "reverse.rm2"
        result_average_rm2 <- average.rm2(value$obs, value$pred)
        names(result_average_rm2) <- "average.rm2"
        result_delta <- delta.rm2(value$obs, value$pred)
        names(result_delta) <- "delta.rm"

        results[[i]] <- c(result, result_rm2, results_reverse, result_average_rm2,
            result_delta)
    }
    return(results)
    stopCluster(cl)
}

mean_and_sd <- function(x) {
    c(round(rowMeans(x, na.rm = TRUE), digits = 2), round(genefilter::rowSds(x,
        na.rm = TRUE), digits = 2))
}

randomForest_test <- function(train, test) {
    ok <- randomForest_testing(train, test)
    data <- data.frame(ok)
    result <- mean_and_sd(data)
    df <- data.frame(result)
    R2_and_RMSE <- t(df)
    label <- c("RMSE_Mean", "Rsquared_Mean", "RM2_Mean", "Reverse_RM2_Mean",
        "Average_RM2_Mean", "Delta_RM2_Mean", "RMSE_SD", "Rsquared_SD", "RM2_SD",
        "Reverse_RM2_SD", "Average_RM2_SD", "Delta_RM2_SD")
    colnames(R2_and_RMSE) <- label
    return(R2_and_RMSE)
}
```

Prediction New AChE Inhibitors from Chembl 21 using Training Data Set from Chembl20

```
training <- readRDS("data.Rds")
testing <- readRDS("Chembl_21.Rds")
```

```r
data_frames <- c("AtomPairs2D_fingerPrintCount", "AtomPairs2D_fingerPrinter",
    "Substructure_fingerPrintCount", "Substructure_fingerPrinter", "Extended_finterPrinter",
    "FingerPrinter", "Estate_FingerPrinter", "GraphOnly_FingerPrinter", "KlekotaRoth_FingerprintCount",
    "KlekotaRoth_FingerPrinter", "MACCS_FingerPrinter", "Pubchem_FingerPrinter")

results <- list()

for (i in data_frames) {
    train <- as.data.frame(training[[i]])
    test <- as.data.frame(testing[[i]])
    result <- randomForest_test(train, test)
    results[[i]] <- result
}
```

Loading required package: foreach

Loading required package: iterators

Loading required package: snow

##
Attaching package: 'snow'

The following objects are masked from 'package:parallel':
##
clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
clusterExport, clusterMap, clusterSplit, makeCluster,
parApply, parCapply, parLapply, parRapply, parSapply,
splitIndices, stopCluster

Warning: closing unused connection 68 (<-localhost:11691)

Warning: closing unused connection 67 (<-localhost:11691)

Warning: closing unused connection 66 (<-localhost:11691)

Warning: closing unused connection 65 (<-localhost:11691)

Warning: closing unused connection 64 (<-localhost:11691)

Warning: closing unused connection 63 (<-localhost:11691)

Warning: closing unused connection 62 (<-localhost:11691)

Warning: closing unused connection 61 (<-localhost:11691)

Warning: closing unused connection 60 (<-localhost:11691)

Warning: closing unused connection 59 (<-localhost:11691)

Warning: closing unused connection 58 (<-localhost:11691)

Warning: closing unused connection 57 (<-localhost:11691)

Warning: closing unused connection 56 (<-localhost:11691)

Warning: closing unused connection 55 (<-localhost:11691)

Warning: closing unused connection 54 (<-localhost:11691)

Warning: closing unused connection 53 (<-localhost:11691)

Warning: closing unused connection 52 (<-localhost:11691)

```
## Warning: closing unused connection 51 (<-localhost:11691)
## Warning: closing unused connection 50 (<-localhost:11691)
## Warning: closing unused connection 49 (<-localhost:11691)
## Warning: closing unused connection 48 (<-localhost:11691)
## Warning: closing unused connection 47 (<-localhost:11691)
## Warning: closing unused connection 46 (<-localhost:11691)
## Warning: closing unused connection 45 (<-localhost:11691)
## Warning: closing unused connection 44 (<-localhost:11691)
## Warning: closing unused connection 43 (<-localhost:11691)
## Warning: closing unused connection 42 (<-localhost:11691)
## Warning: closing unused connection 41 (<-localhost:11691)
## Warning: closing unused connection 40 (<-localhost:11691)
## Warning: closing unused connection 39 (<-localhost:11691)
## Warning: closing unused connection 38 (<-localhost:11691)
## Warning: closing unused connection 37 (<-localhost:11691)
## Warning: closing unused connection 36 (<-localhost:11691)
## Warning: closing unused connection 35 (<-localhost:11691)
## Warning: closing unused connection 34 (<-localhost:11691)
## Warning: closing unused connection 33 (<-localhost:11691)
## Warning: closing unused connection 32 (<-localhost:11691)
## Warning: closing unused connection 31 (<-localhost:11691)
## Warning: closing unused connection 30 (<-localhost:11691)
## Warning: closing unused connection 29 (<-localhost:11691)
## Warning: closing unused connection 28 (<-localhost:11691)
## Warning: closing unused connection 27 (<-localhost:11691)
## Warning: closing unused connection 26 (<-localhost:11691)
## Warning: closing unused connection 25 (<-localhost:11691)
## Warning: closing unused connection 24 (<-localhost:11691)
## Warning: closing unused connection 23 (<-localhost:11691)
## Warning: closing unused connection 22 (<-localhost:11691)
## Warning: closing unused connection 21 (<-localhost:11691)
## Warning: closing unused connection 20 (<-localhost:11691)
## Warning: closing unused connection 19 (<-localhost:11691)
## Warning: closing unused connection 18 (<-localhost:11691)
## Warning: closing unused connection 17 (<-localhost:11691)
## Warning: closing unused connection 16 (<-localhost:11691)
```

```
## Warning: closing unused connection 15 (<-localhost:11691)

## Warning: closing unused connection 14 (<-localhost:11691)

## Warning: closing unused connection 13 (<-localhost:11691)
results

## $AtomPairs2D_fingerPrintCount
##        RMSE_Mean Rsquared_Mean RM2_Mean Reverse_RM2_Mean Average_RM2_Mean
## result      0.95          0.45     0.44             0.08             0.26
##        Delta_RM2_Mean RMSE_SD Rsquared_SD RM2_SD Reverse_RM2_SD
## result           0.36    0.02        0.02   0.02           0.03
##        Average_RM2_SD Delta_RM2_SD
## result           0.02         0.03
##
## $AtomPairs2D_fingerPrinter
##        RMSE_Mean Rsquared_Mean RM2_Mean Reverse_RM2_Mean Average_RM2_Mean
## result      1.01          0.37     0.36             0.01             0.19
##        Delta_RM2_Mean RMSE_SD Rsquared_SD RM2_SD Reverse_RM2_SD
## result           0.35    0.03        0.03   0.04           0.03
##        Average_RM2_SD Delta_RM2_SD
## result           0.03         0.03
##
## $Substructure_fingerPrintCount
##        RMSE_Mean Rsquared_Mean RM2_Mean Reverse_RM2_Mean Average_RM2_Mean
## result      1.13           0.3     0.26             0.04             0.15
##        Delta_RM2_Mean RMSE_SD Rsquared_SD RM2_SD Reverse_RM2_SD
## result           0.22    0.03        0.03   0.03           0.02
##        Average_RM2_SD Delta_RM2_SD
## result           0.02         0.04
##
## $Substructure_fingerPrinter
##        RMSE_Mean Rsquared_Mean RM2_Mean Reverse_RM2_Mean Average_RM2_Mean
## result       1.2          0.27     0.18             0.09             0.13
##        Delta_RM2_Mean RMSE_SD Rsquared_SD RM2_SD Reverse_RM2_SD
## result            0.1    0.03        0.03   0.03           0.02
##        Average_RM2_SD Delta_RM2_SD
## result           0.02         0.02
##
## $Extended_finterPrinter
##        RMSE_Mean Rsquared_Mean RM2_Mean Reverse_RM2_Mean Average_RM2_Mean
## result      0.98          0.45     0.44             0.12             0.28
##        Delta_RM2_Mean RMSE_SD Rsquared_SD RM2_SD Reverse_RM2_SD
## result           0.32    0.02        0.02   0.02           0.03
##        Average_RM2_SD Delta_RM2_SD
## result           0.02         0.04
##
## $FingerPrinter
##        RMSE_Mean Rsquared_Mean RM2_Mean Reverse_RM2_Mean Average_RM2_Mean
## result      0.99          0.45     0.43             0.11             0.27
##        Delta_RM2_Mean RMSE_SD Rsquared_SD RM2_SD Reverse_RM2_SD
## result           0.33    0.02        0.02   0.02           0.03
##        Average_RM2_SD Delta_RM2_SD
## result           0.02         0.03
##
```

```
## $Estate_FingerPrinter
##        RMSE_Mean Rsquared_Mean RM2_Mean Reverse_RM2_Mean Average_RM2_Mean
## result      1.23          0.22     0.16             0.03              0.1
##        Delta_RM2_Mean RMSE_SD Rsquared_SD RM2_SD Reverse_RM2_SD
## result           0.13    0.03        0.03   0.03           0.01
##        Average_RM2_SD Delta_RM2_SD
## result           0.02         0.02
##
## $GraphOnly_FingerPrinter
##        RMSE_Mean Rsquared_Mean RM2_Mean Reverse_RM2_Mean Average_RM2_Mean
## result      1.04          0.34     0.33            -0.02             0.15
##        Delta_RM2_Mean RMSE_SD Rsquared_SD RM2_SD Reverse_RM2_SD
## result           0.35    0.03        0.04   0.04           0.03
##        Average_RM2_SD Delta_RM2_SD
## result           0.03         0.03
##
## $KlekotaRoth_FingerprintCount
##        RMSE_Mean Rsquared_Mean RM2_Mean Reverse_RM2_Mean Average_RM2_Mean
## result      1.03          0.46     0.38             0.23             0.31
##        Delta_RM2_Mean RMSE_SD Rsquared_SD RM2_SD Reverse_RM2_SD
## result           0.16    0.03        0.02   0.03           0.02
##        Average_RM2_SD Delta_RM2_SD
## result           0.02         0.03
##
## $KlekotaRoth_FingerPrinter
##        RMSE_Mean Rsquared_Mean RM2_Mean Reverse_RM2_Mean Average_RM2_Mean
## result      1.07          0.37     0.32             0.09             0.21
##        Delta_RM2_Mean RMSE_SD Rsquared_SD RM2_SD Reverse_RM2_SD
## result           0.23    0.03        0.03   0.03           0.02
##        Average_RM2_SD Delta_RM2_SD
## result           0.02         0.03
##
## $MACCS_FingerPrinter
##        RMSE_Mean Rsquared_Mean RM2_Mean Reverse_RM2_Mean Average_RM2_Mean
## result      1.09          0.36     0.32             0.07              0.2
##        Delta_RM2_Mean RMSE_SD Rsquared_SD RM2_SD Reverse_RM2_SD
## result           0.25    0.03        0.02   0.03           0.03
##        Average_RM2_SD Delta_RM2_SD
## result           0.02         0.03
##
## $Pubchem_FingerPrinter
##        RMSE_Mean Rsquared_Mean RM2_Mean Reverse_RM2_Mean Average_RM2_Mean
## result      0.98          0.45     0.42             0.03             0.23
##        Delta_RM2_Mean RMSE_SD Rsquared_SD RM2_SD Reverse_RM2_SD
## result            0.4    0.03        0.03   0.03           0.04
##        Average_RM2_SD Delta_RM2_SD
## result           0.03         0.02
```