HW Mnist 資料集視覺化

高嘉妤、柯堯珹、吳承恩、趙友誠

2024-11-02

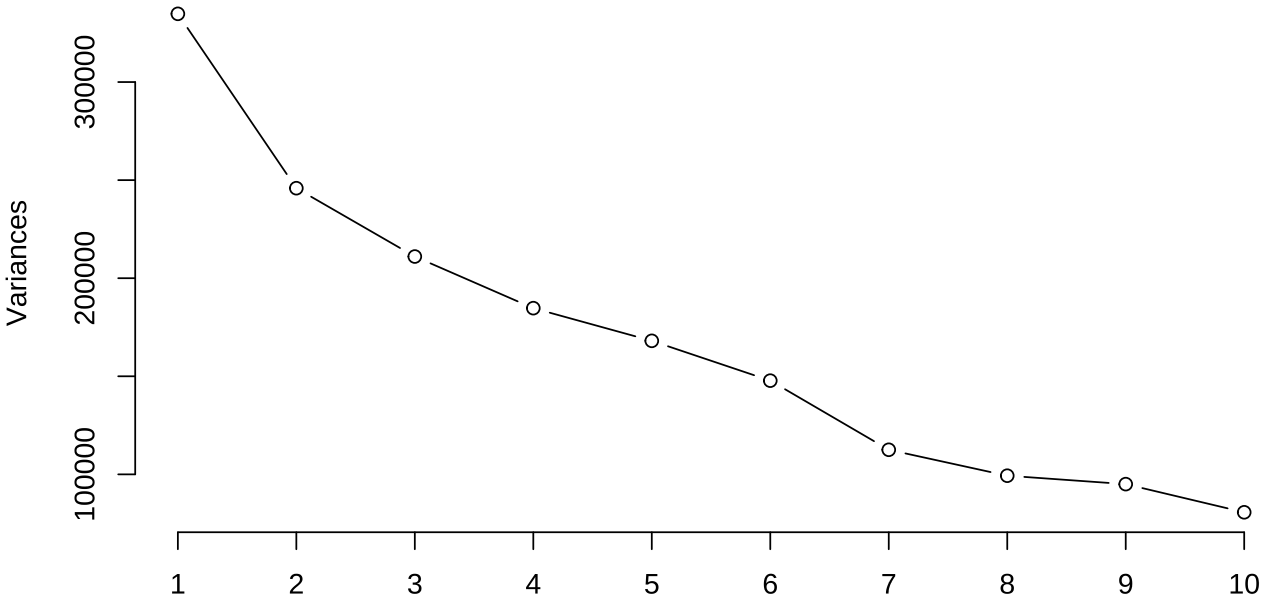
Table of contents


```
mnist <- data.table::fread("MNIST_train.csv")
```

Visualization by PCA

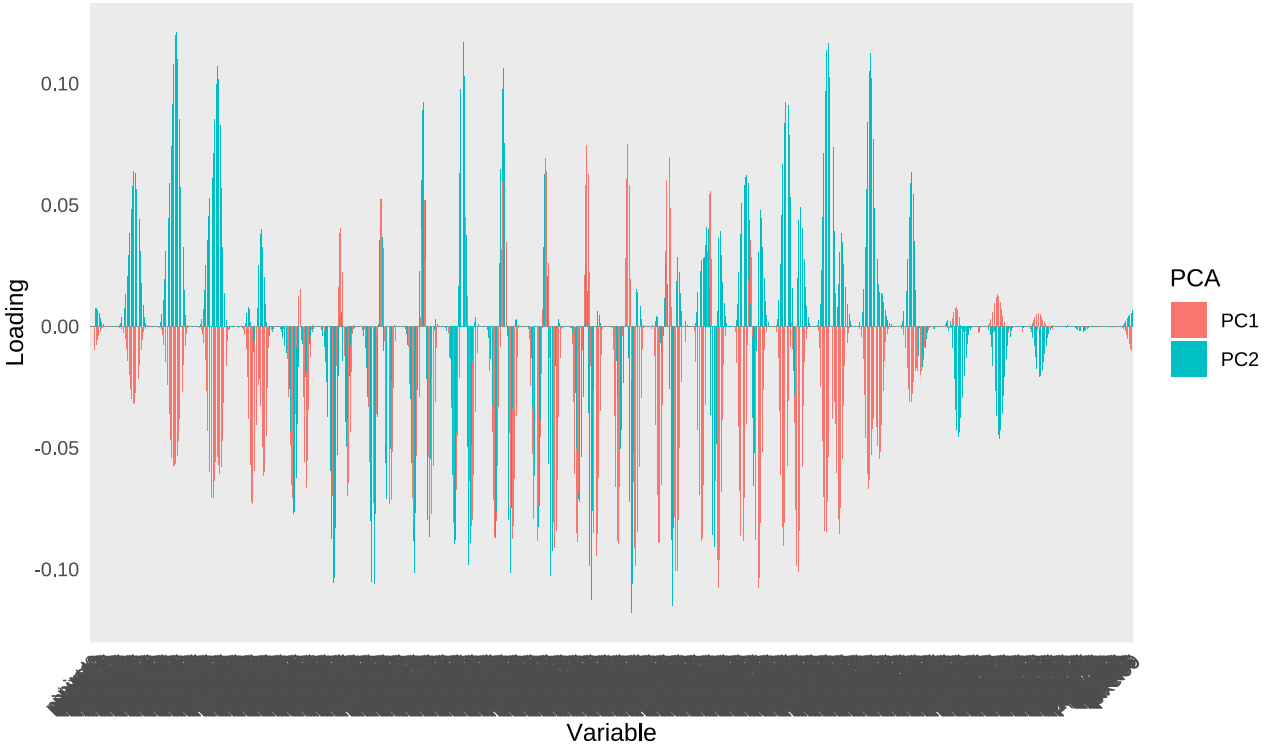
```
library(ggplot2)
library(dplyr)
library(showtext)
showtext_auto()
pca<-prcomp(mnist[,2:785],center = T)
# 做 Scree plot 可以發現 1 到 2 的斜率是最大的，因此我取到 pc2
screeplot(pca,type = 'line', main= "Fig 1: Scree plot of explained variabilities in PCA")
```

Fig 1: Scree plot of explained variabilities in PCA



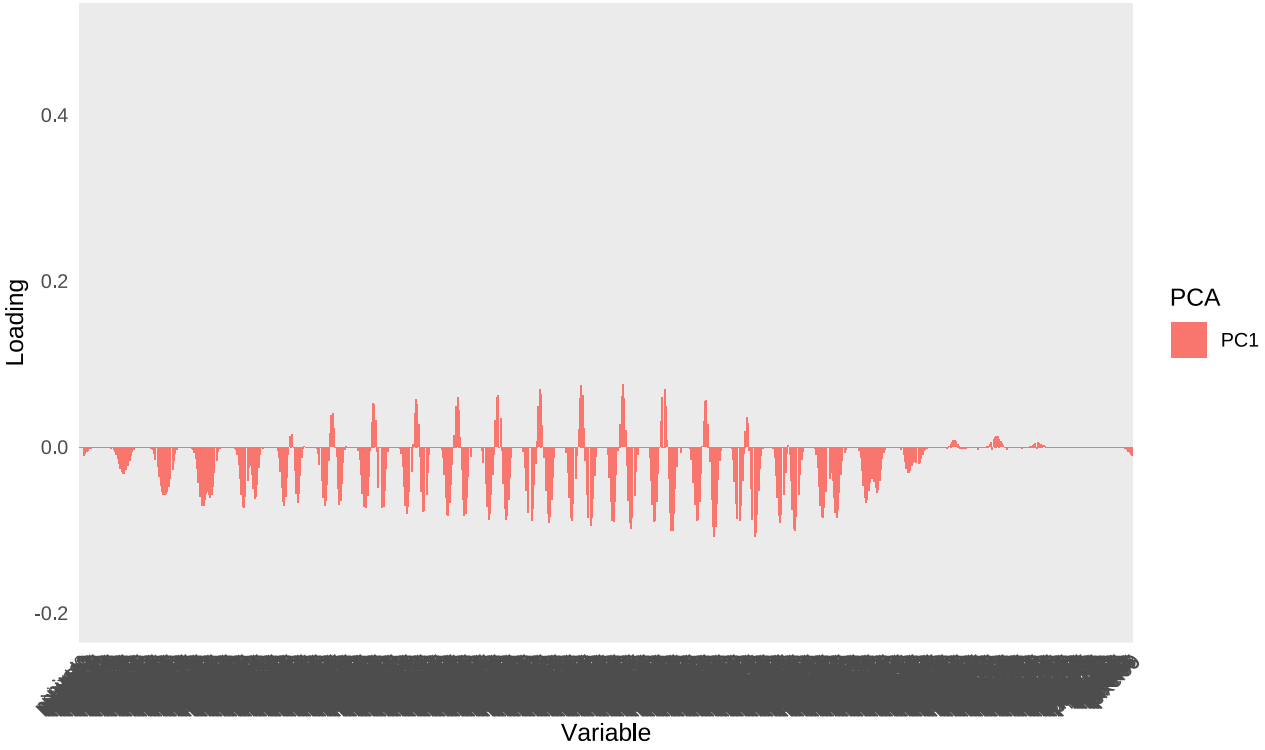

```r
# 取 pc1 跟 pc2
rec_pc <- as.data.frame(pca$x[,1:2])
# 將數字的 label 加進去 pc1 跟 pc2 的 dataset
rec_pc$label<-mnist$label

# 轉成 character 以方便後續做圖
rec_pc$label <- as.character(rec_pc$label)

#pc1 pc2 的 loading (但是變數太多看不出什麼東西)
pca_rotation<-pca$rotation
pca_rotation_df <- data.frame(Variable = rownames(pca_rotation),
                              PC1 = pca_rotation[, 1],
                              PC2 = pca_rotation[, 2])
pca_rotation_longs <- tidyr::pivot_longer(pca_rotation_df,
                                          cols = -Variable,
                                          names_to = "PCA",
                                          values_to = "Rotation")
pca_rotation1<-pca_rotation_longs%>%
  filter(PCA=='PC1')
pca_rotation2<-pca_rotation_longs%>%
  filter(PCA=='PC2')

#loading 圖 pc1 跟 pc2 一起的
ggplot(pca_rotation_longs, aes(x = Variable, y = Rotation, fill = PCA)) +
  geom_bar(stat = "identity", position = position_dodge()) +
  labs(title = "Fig 2: Loadings of PCs", x = "Variable", y = "Loading") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        plot.title = element_text(hjust = 0.5))
```

Fig 2: Loadings of PCs



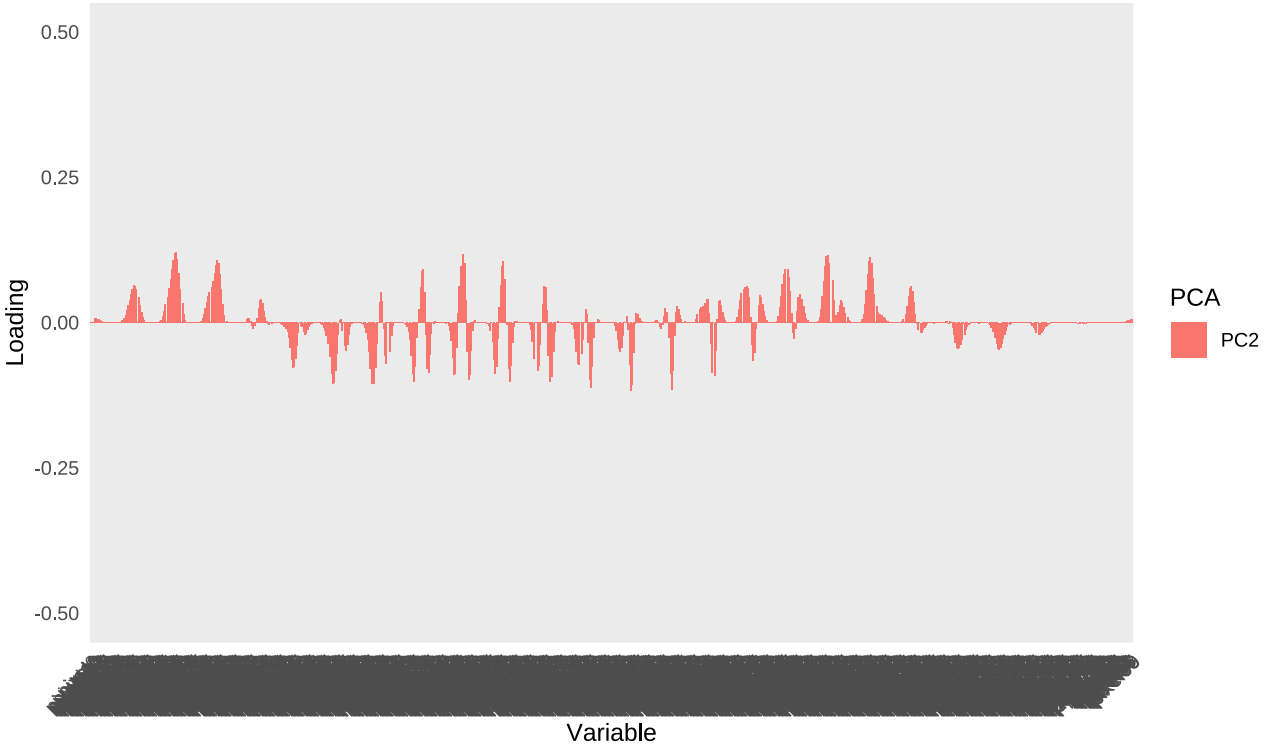

```
# 單獨 pc1 loading 的圖
ggplot(pca_rotation1, aes(x = Variable, y = Rotation, fill = PCA)) +
  geom_bar(stat = "identity", position = position_dodge()) +
  labs(title = "Fig 3: Loadings of PC1", x = "Variable", y = "Loading") +
  theme_minimal() +
  scale_y_continuous(limits = c(-0.2,0.5))+
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        plot.title = element_text(hjust = 0.5))
```

Fig 3: Loadings of PC1



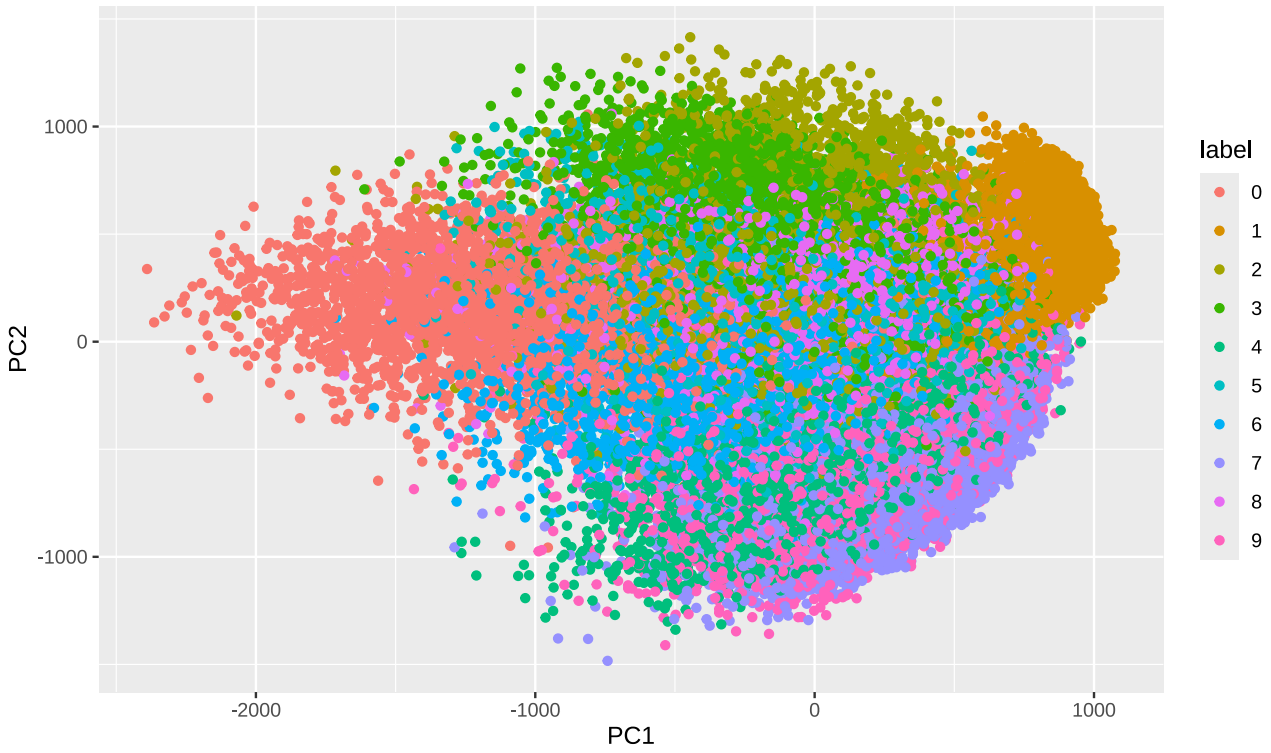

```
# 單獨 pc2 loading 的圖
ggplot(pca_rotation2, aes(x = Variable, y = Rotation, fill = PCA)) +
  geom_bar(stat = "identity", position = position_dodge()) +
  labs(title = "Fig 4: Loadings of PC2", x = "Variable", y = "Loading") +
  theme_minimal() +
  scale_y_continuous(limits = c(-0.5,0.5))+
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        plot.title = element_text(hjust = 0.5))
```

Fig 4: Loadings of PC2




```
# 製作 pc1 對 pc2 的圖，也可以看出大部份同一個 label 都還是有 cluster 形成一個 group
ggplot(rec_pc,aes(x = PC1, y = PC2,colour = label))+
  geom_point()+ labs(title = 'Fig 5: PC1 vs. PC2')+
  theme(plot.title = element_text(hjust = 0.5), legend.position = 'right')
```

Fig 5: PC1 vs. PC2



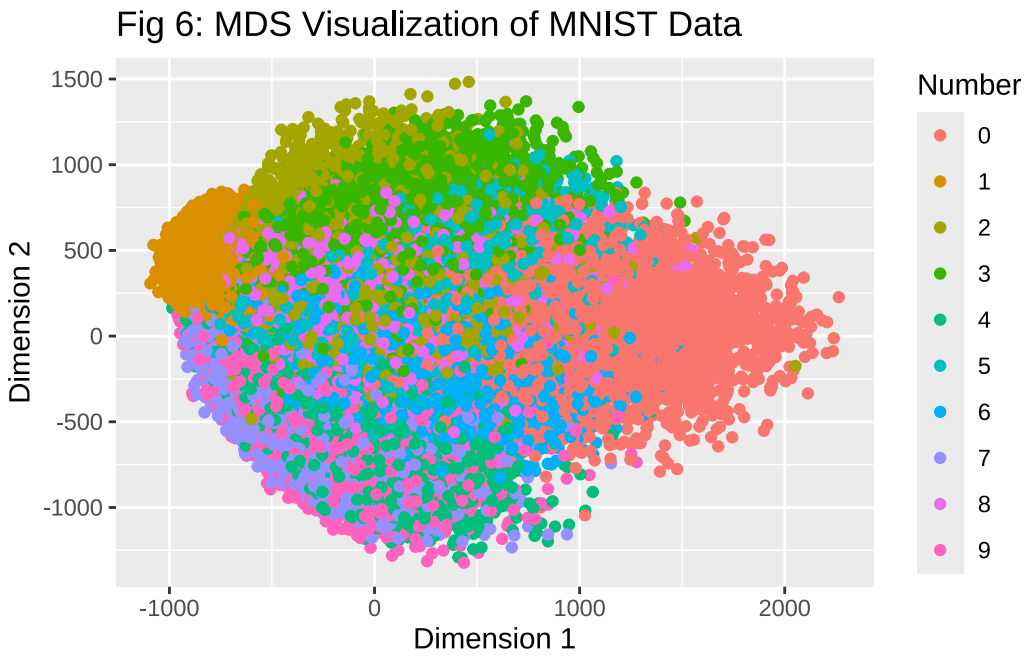
Visualization by MDS

```
library(Rcpp)
cppFunction('
  NumericMatrix calculate_2distance(NumericMatrix X, NumericMatrix X_sample) {
    int r1 = X.nrow();
    int r2 = X_sample.nrow();
    int ncol = X.ncol();
    NumericMatrix D(r1, r2);
    for (int i = 0; i < r1; ++i) {
        for (int j = 0; j < r2; ++j) {
            float entry_ij = 0.0;
            for (int k = 0; k < ncol; ++k) {
                entry_ij += (X(i, k) - X_sample(j, k)) * (X(i, k) - X_sample(j, k));
            }
            D(i, j) = sqrt(entry_ij);
        }
    }
    return D;
}
')
# 後來沒有使用，但如果想計算全部資料的 MDS 會使用到
cppFunction('
  NumericMatrix calculate_B(NumericMatrix mat_dist, int n) {
    NumericMatrix B(n, n);
    NumericVector row_sum(n);
    NumericVector col_sum(n);
```

```
    double total_sum = 0;

    for (int i = 0; i < n; ++i) {
      for (int j = 0; j < n; ++j) {
        double value = mat_dist(i, j) * mat_dist(i, j);
        total_sum += value;
        row_sum[i] += value;
        col_sum[j] += value;
      }
    }

    for (int i = 0; i < n; ++i) {
      for (int j = 0; j < n; ++j) {
        B(i, j) = -0.5 * (mat_dist(i, j) * mat_dist(i, j) - row_sum[i] / n - col_sum[j] / n + total_sum
      }
    }
    return B;
  }
')

cppFunction('
  NumericMatrix calculate_distance(NumericMatrix mat_data) {
    int nrow = mat_data.nrow();
    int ncol = mat_data.ncol();
    NumericMatrix D(nrow, nrow);
    for (int i = 0; i < nrow; ++i) {
        for (int j = i + 1; j < nrow; ++j) {
            float entry_ij = 0.0;
            for (int k = 0; k < ncol; ++k) {
                entry_ij += (mat_data(i, k) - mat_data(j, k)) * (mat_data(i, k) - mat_data(j, k));
            }
            D(i, j) = sqrt(entry_ij);
            D(j, i) = D(i, j);
        }
    }
    return D;
}
')
```

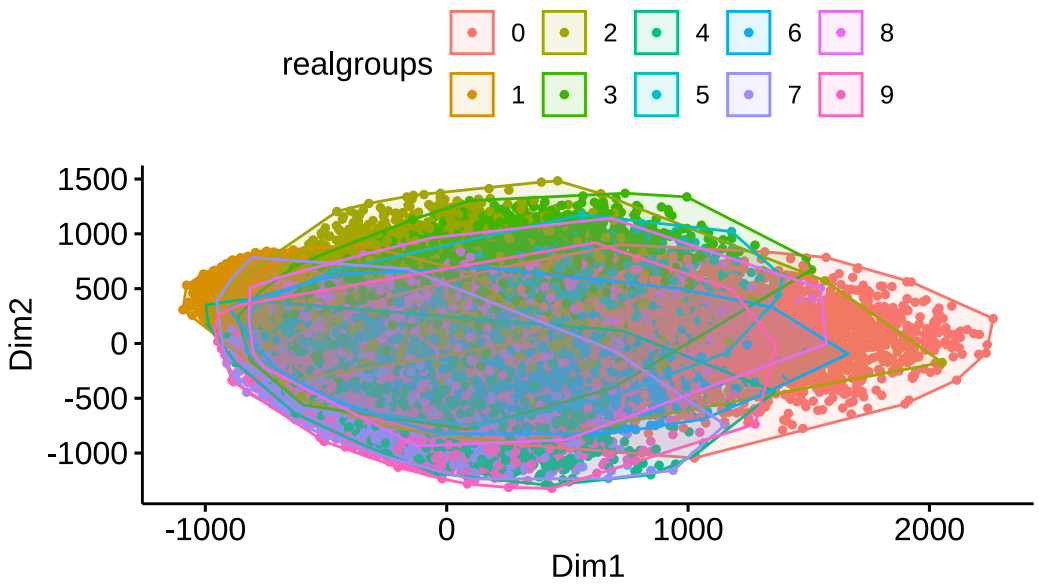
```
library(magrittr)
library(dplyr)
library(ggpubr)
library(proxy)
library(parallelDist)
# 用 mds 方法將將點與點的距離降維
subset_mds <- function(mnist, sample_size = 1000) {
  X <- as.matrix(mnist[,2:785])
  n <- nrow(X)

  # 隨機選擇子集
  set.seed(1)
  idx <- sample(n, sample_size)
  X_sample <- X[idx,]
```

```
  # 在子集上計算 MDS
  d_sample <- calculate_distance(X_sample)
  mds_sample <- cmdscale(d_sample)

  # 使用 Nyström 方法擴展到所有點
  K_sample <- as.matrix(calculate_distance(X_sample))
  K_full <- as.matrix(calculate_2distance(X, X_sample))

  mds <- K_full %*% solve(K_sample + diag(1e-6, nrow(K_sample))) %*% mds_sample %>%
    as_tibble()

  return(mds)
}
mds = subset_mds(mnist)
colnames(mds) <-c('Dim1','Dim2')
ggplot(data = mds)+
  geom_point(aes(x=Dim1, y=Dim2, color = as.factor(mnist$label)))+
  labs(title = "Fig 6: MDS Visualization of MNIST Data",
       x = "Dimension 1",
       y = "Dimension 2",
       color = "Number") +
  theme(legend.position="right")
```

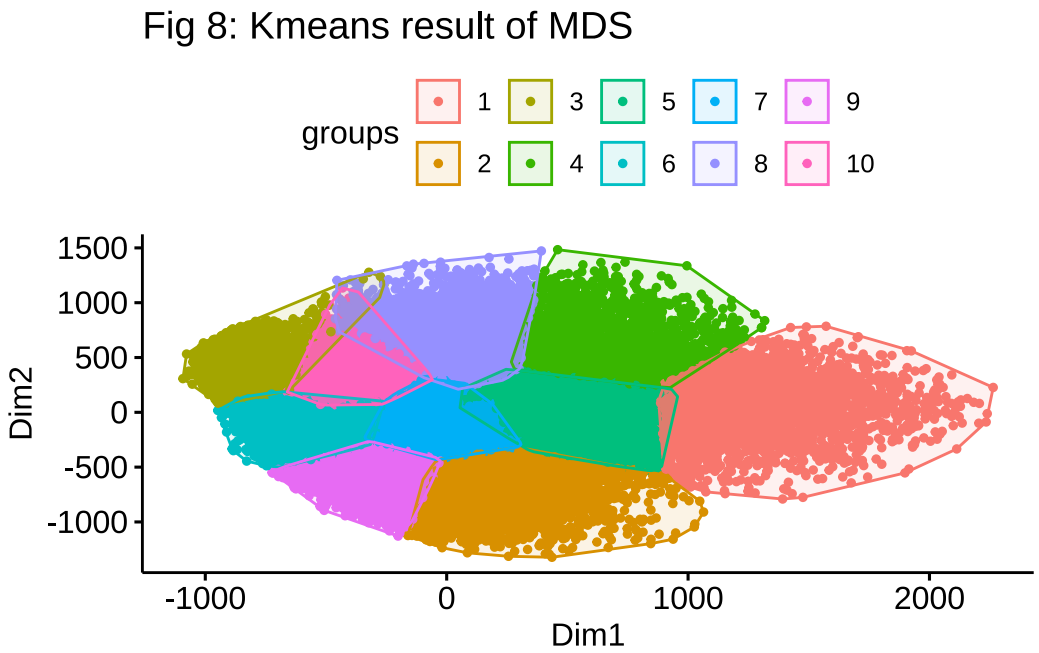



```
clust <- kmeans(mds,10)$cluster %>% as.factor()
mds <- mds %>% mutate(groups=clust)
mds <- mds %>% mutate(realgroups=as.factor(mnist$label))

# 圖片為每個點在第一維度對上第二維度的散佈圖
# 可以發現同一群還是大致上有聚集再一起，並且跟 pca 呈現的圖片類似
ggscatter(mds,x="Dim1",y="Dim2",
          size=1, rapel=TRUE,
          color="realgroups",
          ellipse=TRUE,
```

```
          ellipse.type = "convex", title="Fig 7: MDS result with true labels")
```




```
ggscatter(mds,x="Dim1",y="Dim2",
          size=1,rapel=TRUE,
          color="groups",
          ellipse=TRUE,
          ellipse.type = "convex",title = "Fig 8: Kmeans result of MDS")
```



計算全部資料點的 **MDS** 的程式碼，全跑會很久

```
library(RSpectra)
library(compiler)
```

```r
library(Matrix)
mat.dist <- calculate_distance(as.matrix(mnist[,-1]))
#Ram 會炸掉 QQ
#mat.dist <- parDist(as.matrix(mnist[1:2000,-1]),method = "euclidean")
myMDS.2D <- cmpfun(function(mat.dist){
  n <- nrow(mat.dist)
  B <- calculate_B(mat.dist, n)
  eigen_decomp <- eigs_sym(B, k=2 ,which = "LM")
  return(eigen_decomp$vectors %*% diag(sqrt(eigen_decomp$values)))
})
mds_result <- data.frame(myMDS.2D(mat.dist))
ggplot(data = mds_result)+
  geom_point(aes(x=X1, y=X2, color = as.factor(mnist$label)))+
  labs(title = "Fig 9: MDS Visualization",
       x = "Dimension 1",
       y = "Dimension 2",
       color = "Number")
```

Visualization by t-SNE

```r
library(Rtsne)
set.seed(123)
tsne_result <- Rtsne(mnist[,-1], dims = 2)
tsne_df <- as.data.frame(tsne_result$Y)
# 計算數字要顯示在每一群的中心點
centroids <- tsne_df %>%
  mutate(label = mnist$label) %>%
  group_by(label) %>%
  summarize(V1 = mean(V1), V2 = mean(V2), .groups = 'drop')

ggplot(data = tsne_df)+
  geom_point(aes(x=V1, y=V2, color  = as.factor(mnist$label)))+
  labs(title = "Fig 9: t-SNE Visualization of MNIST Data",
       x = "Dimension 1",
       y = "Dimension 2",
       color = "Number") +
  geom_text(data = centroids, aes(x = V1, y = V2, label = label),
            vjust = -1, size = 4, color = "black")+
  theme(legend.position = "right")
```

Fig 9: t-SNE Visualization of MNIST Data