https://datascienceplus.com/how-to-perform-logistic-regression-lda-qda-in-r/
https://rstudio-pubs-static.s3.amazonaws.com/336635_7611ceab3e324623b9a7bea8de2b3818.html
---

```
# mean parameters of the k class
pi_lda <- function(y){
  pi_est <- table(y) / length(y)
  return(as.matrix(pi_est))
}


# Output the group means
# these are the average of the predictor within each class,
# and are used by LDA as estimates of mu k
mu_lda <- function(X, y){
  data_est <- as.data.frame(cbind(X,y))
  data_est$X <- as.numeric(as.character(data_est$X))
  mu <- aggregate(data = data_est, X ~ y, FUN = "mean")
  colnames(mu) <- c("y", "X")
  return(mu)
}

# variance parameter of the k class
var_lda <- function(X, y, mu){
  n <- length(X)
  K <- length(unique(y))
  k <- unique(y)
  var_est <- 0

  for (i in 1:K){
    var_est <- sum((X[y == k[i]] - mu$X[k[i] == mu$y])^2) + var_est
  }

  var_est <- (1 / (n - K)) * var_est
  return(var_est)
}


# discriminant function for p = 1
discriminant_lda <- function(X, pi, mu, var){
  K <- length(unique(y))
  k <- unique(y)

  disc <- matrix(nrow = length(X), ncol = K)
  colnames(disc) <- k

  for (i in 1:K){
```

```
    disc[ ,i] <- X * (mu$X[i] / var) - ((mu$X[i]^2) / (2 * var)) +
log(pi[i])
  }

  disc <- as.data.frame(disc)
  disc$predict <- apply(disc, 1, FUN = "which.max")
  return(disc)
}
########test
X <- iris[ ,1]
y <- as.character(iris[ ,5])

pi_est <- pi_lda(y)
mu_est <- mu_lda(X, y)
var_est <- var_lda(X, y, mu_est)
discriminant_est <- discriminant_lda(X, pi_est, mu_est, var_est)

table(discriminant_est$predict, iris$Species)
```