

HW1_r

2026-01-23

```
# read in data
data <- read.csv("evs.csv", stringsAsFactors = TRUE)

# one hot encode data
onehot_data <- model.matrix(~ ., data = data)[, -1]

onehot_data <- as.data.frame(onehot_data)
head(onehot_data)
```

```
##      id driveRear acceleration topspeed electricrange totalpower batterycapacity
## 1 1647         1         7.8      185         390        168         77.4
## 2 1252         1         5.7      190         470        250         83.9
## 3 1534         1         7.9      160         450        150         82.0
## 4 1178         1         8.9      160         275        110         55.0
## 5 1279         1         8.7      160         330        132         62.0
## 6 1280         1         8.6      160         420        150         82.0
##   chargespeed
## 1          49
## 2          54
## 3          55
## 4          38
## 5          53
## 6          51
```

```
# convert data to matrix
numpy_data <- as.matrix(onehot_data)
dim(numpy_data)
```

```
## [1] 119  8
```

```
# separate X and y features
y <- numpy_data[, 2]
n_cols <- ncol(numpy_data)
X <- numpy_data[, 3:n_cols]
print(dim(X))
```

```
## [1] 119  6
```

```
print(length(y))
```

```
## [1] 119
```

```
# min-max scale the data
```

```
library(scales)
```

```
X_scale <- apply(X, 2, function(x) rescale(x, to = c(0, 1)))
```

```
head(X_scale)
```

```
##      acceleration  topspeed electricrange totalpower batterycapacity chargespeed
## 1  0.17518248 0.5714286      0.5049505  0.4258675      0.5571726  0.2906977
## 2  0.02189781 0.6190476      0.6633663  0.6845426      0.6247401  0.3488372
## 3  0.18248175 0.3333333      0.6237624  0.3690852      0.6049896  0.3604651
## 4  0.25547445 0.3333333      0.2772277  0.2429022      0.3243243  0.1627907
## 5  0.24087591 0.3333333      0.3861386  0.3123028      0.3970894  0.3372093
## 6  0.23357664 0.3333333      0.5643564  0.3690852      0.6049896  0.3139535
```

```
# add bias term
```

```
X_bias <- cbind(1, X_scale)
```

```
dim(X_bias)
```

```
## [1] 119  7
```

```
# train/test split
```

```
set.seed(42)
```

```
train_size <- floor(0.7 * nrow(X_bias))
```

```
train_idx <- sample(seq_len(nrow(X_bias)), size = train_size)
```

```
X_train <- X_bias[train_idx, ]
```

```
X_test  <- X_bias[-train_idx, ]
```

```
y_train <- y[train_idx]
```

```
y_test  <- y[-train_idx]
```

```
cat("(", dim(X_train), ")", "(", dim(X_test), ")", "(", length(y_train), "),", "(", length(y_test), "),")
```

```
## ( 83 7 ), ( 36 7 ), ( 83 ,), ( 36 ,)
```