DSI – DASI Aug 9, 2023 Brown Bag – Online Discussion on AI
Activity for Advanced Predictive Modeling w R Course

Kellie Keeling (kkeeling@du.edu)

When I started seeing lots of verbose documentation – I knew I was seeing ChatGPT output!

Ask ChatGPT:

```
Suppose I have a dataset created with
df <- tibble(ISLR::Carseats)  %>%
            mutate(SalesHigh = factor(if_else(Sales <= 8, "No", "Yes"),
                                      levels=c("Yes","No"))) %>%
            select(-Sales)
And also a training and test dataset
carseats_split <- initial_split(df, prop = .75, strata=SalesHigh)
carseat_train <- rsample::training(carseats_split)
carseat_test <- rsample::testing(carseats_split)
```

What is the code to create a recipe to perform step_dummy and step_zv?

Overall produced well-documented code, but some Confusion Created

- `step_dummy(all_nominal(), one_hot = TRUE)`
  - Also created dummies for Y variable
  - One_hot = TRUE creates dummies for all (not –1)
- `step_medianimpute()` instead of `step_zv`
- Use of `juice()` (deprecated) instead of `bake()`
- Put `prep()` in different location than my examples
- Did not give all packages needed to run code

Sample output:

```r
library(tidymodels)

# Create a recipe
carseat_recipe <- recipe(SalesHigh ~ ., data = carseat_train) %>%
  step_dummy(all_nominal(), one_hot = TRUE) %>%
  step_zv(all_predictors())

# Prepare the training data using the recipe
carseat_train_preprocessed <- carseat_recipe %>%
  prep() %>%
  bake(new_data = NULL)

# Prepare the test data using the same recipe
carseat_test_preprocessed <- carseat_recipe %>%
  prep() %>%
  bake(new_data = carseat_test)

# Check the preprocessed data
head(carseat_train_preprocessed)
head(carseat_test_preprocessed)
```

First Ask: How can we fit a polynomial kernel with `degree` 2 and `scale factor` = .1 predicting SalesHigh
Second Ask: Can you use tidymodels to answer the previous question?

- Most output did not use tidymodels in first question, but one did.
- Saw two different functions used in non-tidymodels – svm and ksvm
- Not all had the packages loaded
- Is kernelPolynomial actually real? Couldn't find in documentation
- In class we focused on using augment() instead of predict() so I noticed a lot of predict() code showing up in class the past few weeks. Also used confusion_matrix() and I don't think that is real – should be conf_mat().

Samples from the two packages (kernlab and e1071) in most of the results:

```
library(kernlab)
polynomial_kernel <- kernelPolynomial(degree = 2, scale_factor = 0.1)

# Fit a SVM model with the polynomial kernel
svm_model <- ksvm(SalesHigh ~ ., data =
as.data.frame(predictor_matrix), kernel = polynomial_kernel)
```

```
# Load the required package
library(e1071)

# Fit a polynomial kernel SVC
polynomial_svc <- svm(SalesHigh ~ ., data = carseat_train_baked,
                      kernel = "polydot", degree = 2, scale = 0.1)

# Make predictions on the test set
predictions <- predict(polynomial_svc, newdata = carseat_test_baked)

# Evaluate the model's performance
conf_matrix <- table(predictions, carseat_test_baked$SalesHigh)
accuracy <- sum(diag(conf_matrix)) / sum(conf_matrix)

print(conf_matrix)
print(paste("Accuracy:", accuracy))
```

Tidymodels versions were pretty close to class examples except use of predict() instead of augment()

```
library(tidymodels)

# Fit a polynomial kernel SVC using tidymodels
polynomial_svc <- svm_poly(degree = 2, scale = 0.1) %>%
  set_engine("kernlab") %>%
  set_mode("classification") %>%
  fit(SalesHigh ~ ., data = carseat_train_baked)

# Make predictions on the test set
predictions <- predict(polynomial_svc, new_data = carseat_test_baked)

# Evaluate the model's performance
conf_matrix <- confusion_matrix(predictions, truth = carseat_test_baked$Sales
accuracy <- accuracy(conf_matrix)

print(conf_matrix)
print(paste("Accuracy:", accuracy))
```