K-nearest neighbors classification: Customer churn

In this assignment you are going to train a k-nearest neighbors classification model to predict whether a customer of a business is going to cancel their subscription (churn) or not. The data you have contains much information about the customer's subscription, payment method, etc.

This data set is featured in *https://www.kaggle.com/datasets/blastchar/telco-customer-churn* and can be accessed in JASP via *Data Library → 10. Machine Learning → Telco Customer Churn* (click the icon without the logo).

1. Load the data set and open the k-nearest neighbors classification analysis via the Machine Learning menu.

2. Navigate to the training parameters section in the interface and set the seed to 1 to be able to reproduce your results. What effect does this have for your analysis?

First, we are going to try out a couple of values for *k* and see how this affects the accuracy of model predictions on the test set.

3. Set the number of nearest neighbors to "Fixed" and set the value to 1. Enter the variable *Churn* as the target variable and all other variable (except *customerID*) as the features. How many rows of the data are used as the training set? And how many rows are used as the test set?

4. What is the accuracy of the model predictions on the test set? Use the confusion matrix to find out how many observations from the test set are predicted correctly. Can you determine the accuracy of the model given the confusion matrix?

5. Change the number of nearest neighbors used in the model to 3 and find out if this improved the accuracy of the model predictions on the test set.

6. Change the number of nearest neighbors used in the model to 5 and find out if this improved the accuracy of the model predictions on the test set.

Manually trying every value of *k* to find the best accuracy can be a bit tedious, so we can ask JASP to do this for us automatically.

7. Go to the *Training parameters* section and change the number of nearest neighbors from *Fixed* to *Optimized*. How many rows does the test set now contain? And the validation set?

8. What is the best number of *k* and what is the accuracy of this models' predictions on the test set? What is the accuracy of the predictions on the validation set?

9. Create a figure of the optimization by clicking *Classification accuracy*. What does this figure tell you about the optimal value of *k*?

10. Click the *evaluation metrics* table. Which class is the easiest to predict under the optimized model?

11. Change the maximum number of nearest neighbors to 25. Does expanding the search region improve the performance of the best model?

12. Save the model and open the dataset churn_prediction.csv

13. Make predictions for the 100 new customers in the data.

14. How many customers are predicted to leave?