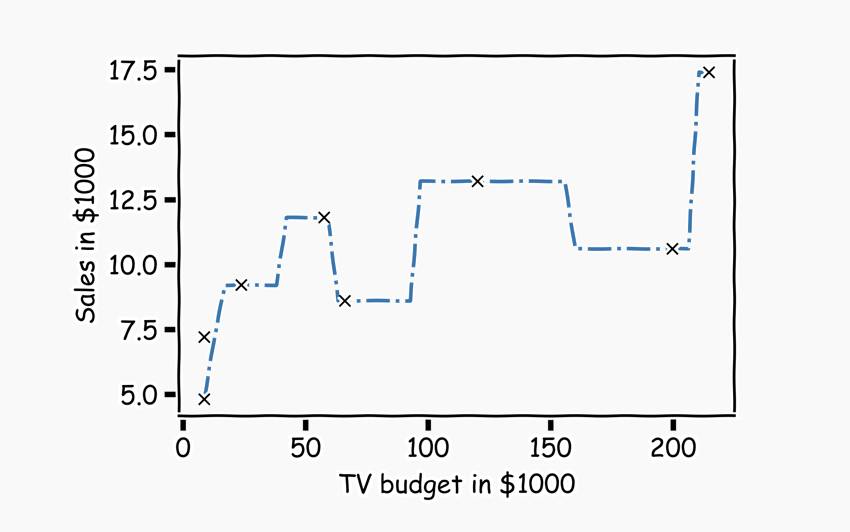
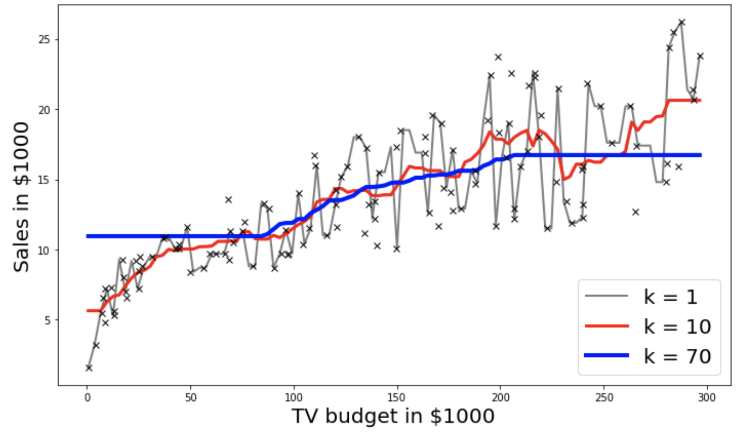
Exercise: Simple kNN Regression

The goal of this exercise is to **re-create the plots** given below. You would have come across these graphs in the lecture as well.





Instructions:

Part 1: KNN by hand for k=1

* Read the Advertisement data.
* Get a subset of the data from row 5 to row 13.
* Apply the kNN algorithm by hand and plot the first graph as given above.

Part 2: Using sklearn package

* Read the Advertisement dataset.
* Split the data into train and test sets using the train\_test\_split() function.
* Set k\_list as the possible k values ranging from 1 to 70.
* For each value of k in k\_list:
  + Use sklearn KNearestNeighbors() to fit train data.
  + Predict on the test data.
  + Use the helper code to get the second plot above for k=1,10,70.

Hints:

**The following are direct links to documentation for some of the functions used in this exercise. As always, if you are unsure how to use a function, refer to its documentation.**

[**np.argsort()**](https://numpy.org/doc/stable/reference/generated/numpy.argsort.html)

Returns the indices that would sort an array.

[**df.iloc[]**](https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.iloc.html)

Returns a subset of the dataframe that is contained in the column range passed as the argument.

[**plt.plot( )**](https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.plot.html)

Plot y versus x as lines and/or markers.

[**df.values**](https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.values.html)

Returns a Numpy representation of the DataFrame.

[**pd.idxmin()**](https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.idxmin.html)

Returns index of the first occurrence of minimum over requested axis.

[**np.min()**](http://pageperso.lif.univ-mrs.fr/~francois.denis/IAAM1/numpy-html-1.14.0/reference/generated/numpy.ndarray.min.html)

Returns the minimum along a given axis.

[**np.max()**](https://numpy.org/doc/stable/reference/generated/numpy.ndarray.max.html)

Returns the maximum along a given axis.

[**model.fit( )**](https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html#sklearn.neighbors.KNeighborsRegressor.fit)

Fit the k-nearest neighbors regressor from the training dataset.

[**model.predict( )**](https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html#sklearn.neighbors.KNeighborsRegressor.predict)

Predict the target for the provided data.

[**np.zeros()**](https://numpy.org/devdocs/reference/generated/numpy.zeros.html)

Returns a new array of given shape and type, filled with zeros.

[**train\_test\_split(X,y)**](http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html)

Split arrays or matrices into random train and test subsets.

[**np.linspace()**](https://numpy.org/doc/stable/reference/generated/numpy.linspace.html)

Returns evenly spaced numbers over a specified interval.

[**KNeighborsRegressor(n\_neighbors=k\_value)**](https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html)

Regression-based on k-nearest neighbors.