Capstone project | Made by Juan David Pisco Jaimes

Clothing size predictor - Desktop app

Project Overview

Predict your clothing size according to your height, age, and weight in an intuitive Qt desktop app (Windows and Mac available) with the help of artificial intelligence to make your online purchases more secure and confident.

This project is a classification ML problem in the fashion field, which had to use data from the Kaggle (datasets) platform.

Problem Statement

Buying clothes online is a headache for all people who are not sure about their size in a certain store, that's why with this app, the customer can be more confident with an ML solution that recommends the clothing size that best fits the customer’s body shape. The idea behind this is to sell the product to clothing stores and reinforce the training with data from new customers so the model is more precise and related to each store.

Metrics

The metrics used for checking if the model was doing right were:

- Cross-Entropy Loss: This is a measure of the difference between two probability distributions for a given random variable or set of events. The formula of CEL is (Given the probabilities of the events P and Q):

  H(P, Q) = – sum x in X P(x) * log(Q(x))

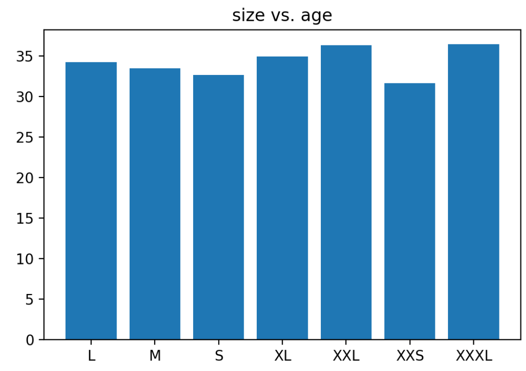
- R^2 score: Although this is a regression metric, it works here for calculating the difference between the predicted values and the real values on a big basis, since if the class predicted was 3 but the real one was 4, this metric will let us know with a maximum value of 1 if the prediction was or not close. R^2 score formula is:

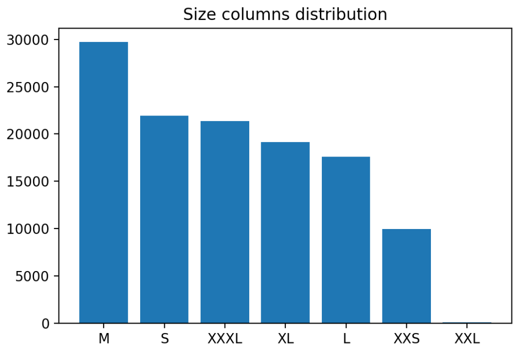
$$R^2 = 1 - \frac{\Sigma(y_i - \widehat{y})^2}{\Sigma(y_i - \widehat{y})^2}$$

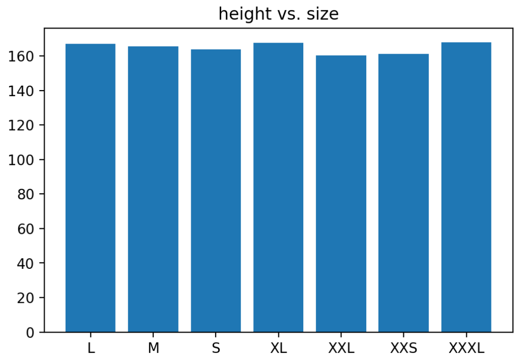
Analysis

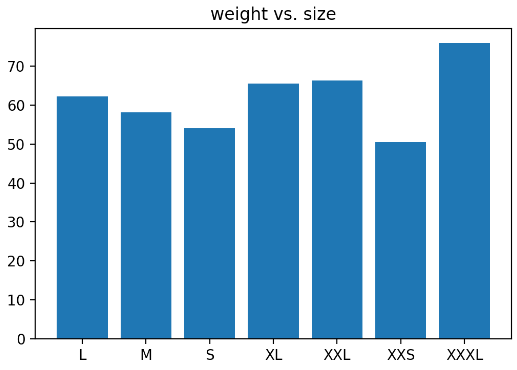
After exploring and plotting results, these were the results of the most important facts about the data collected from this Kaggle post.

- The "final_test.csv" dataset has weight, age, height, and clothing size as columns, in which the first 3 were selected for training the model.
- The number of data points in the whole dataset is: 119734
- The number of columns in the dataset is: 4
- From the data, we can get that there is a bias problem where the range of data is very restricted which can lead to a bad performance of the model, as the data is not varied.



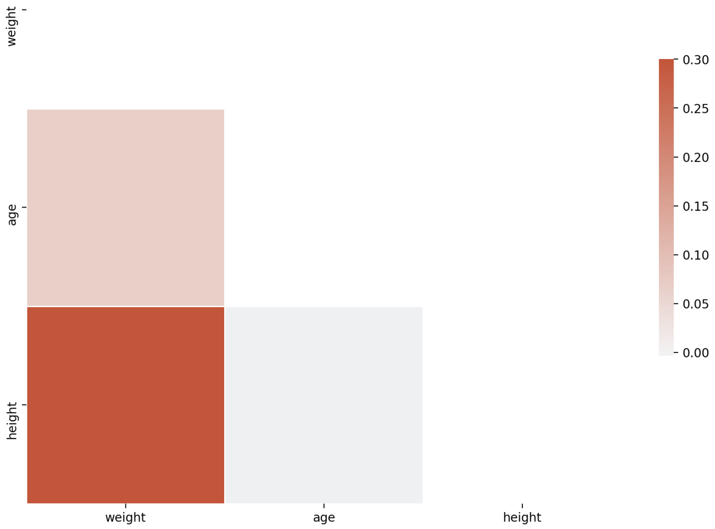






These are the plots of the mean of x variable according to each clothing size. The problem here is the fact that the values are very close to each other and the height-size together with age-size plots don't look coherent at all.

- Concerning correlation between variables, these were the results:



Here, we can see that age with height are not correlated at all and weight with age neither, however, we can't discard this variable as it can lead us to better performance in the model as there are more characteristics to learn, thus the model would be more precise.

Methodology

Data Preprocessing did: The size column was converted to values from 0-6 as Strings cannot be output by a NN (Creating one-hot encoding | dummy variables for the size column could be another way to solve this problem). All weight, age, and height data points were normalized from 0 to 1 according to the maximum and minimum values in each set of column values.

Implementation: Using Pytorch (Facebook's ML framework), a multi-perceptron neural network is created for predicting clothing size. Architecture and dimensional problems in the tensors used were the most problematic factors when creating the model.

Refinement: Parameters such as Learning Rate, number of neurons in the hidden layers, activation functions, and epochs were tuned according to the model's behavior.

Desktop app development: A Qt desktop app was developed for a better UI/UX experience, OOP concepts such as inheritance, instances, methods, and objects were worked through the development together with AdobeXD design for getting a general view of the app.

Results

Desktop app screenshots: *

Model evaluation and validation: The Neural Network model had the following parameters.

- Learning Rate: 0.01
- Optimizer: SGD with momentum
- Number of epochs: 50
- Activation funtions: Relu
- Number of hidden layers: 2
- Number of neurons per hidden layer: 64
- Loss function (Main metric): CrossEntropyLoss (used for classification problems)
- Accuracy metric: R2 Score
- Train-Validation-Test split: 60|20|20
- Validation every 5 batches of training.

Training results of the model were:

- Validation loss (Cross-Entropy) | 1.127
- R2 Score | 67.72

Justification: From the training results we can see that the model didn't perform very well because of the factors mentioned before:

- Variety in data is poor.
- The correlation between data is low.

Conclusion

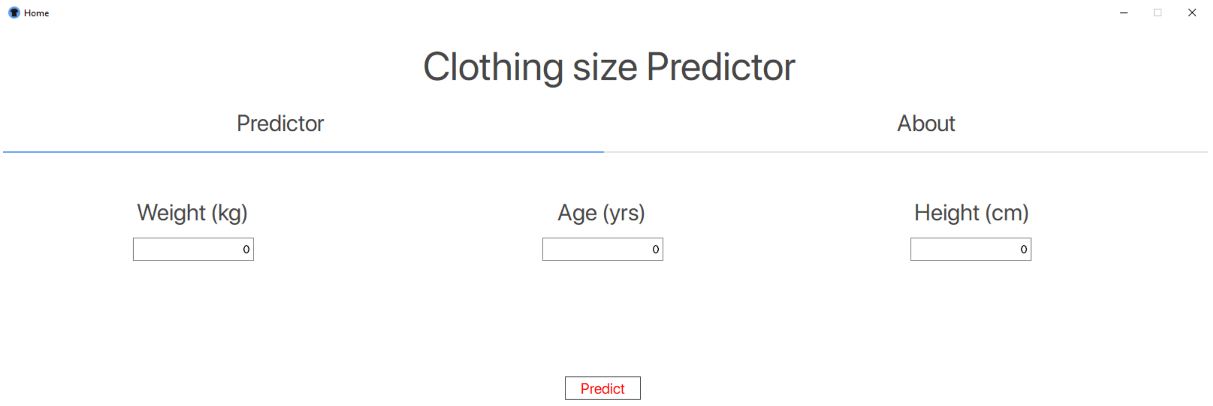
Reflection and Improvements:

For this problem, having more variety of data-points (e.g more height values from kids or ancient people | Having more weight values from kids and overweighted people) would help a lot without thinking about adding the number of variables which would make the model even 2x better however it makes the model and the input values more complex too. Even though the model had an R2 score of 0.6772 (67.72%) on a general basis the model performs decently in the app.

The usability and design of the desktop app can be improved using better design resources for buttons, text fields, etc.

Thanks for reading! ;) | Made by Juan David Pisco

*

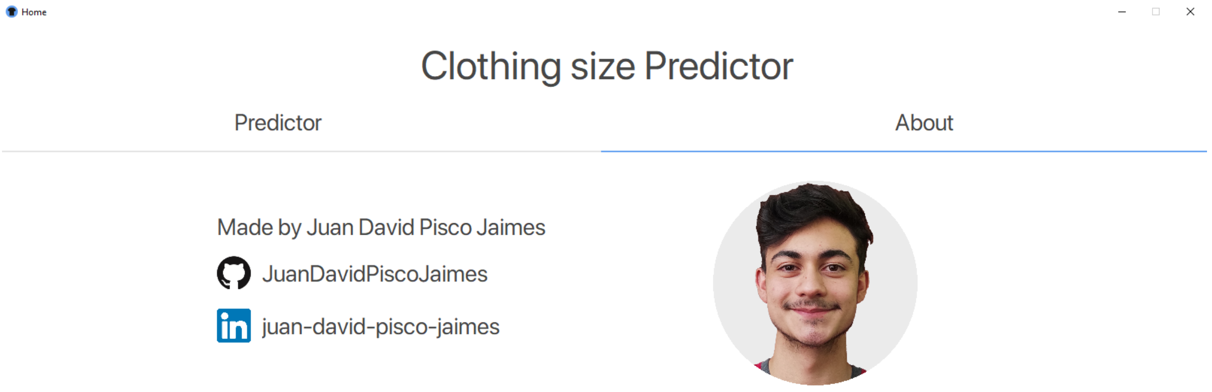


Clothing size Predictor

Weight (kg): 0

Age (yrs): 0

Height (cm): 0

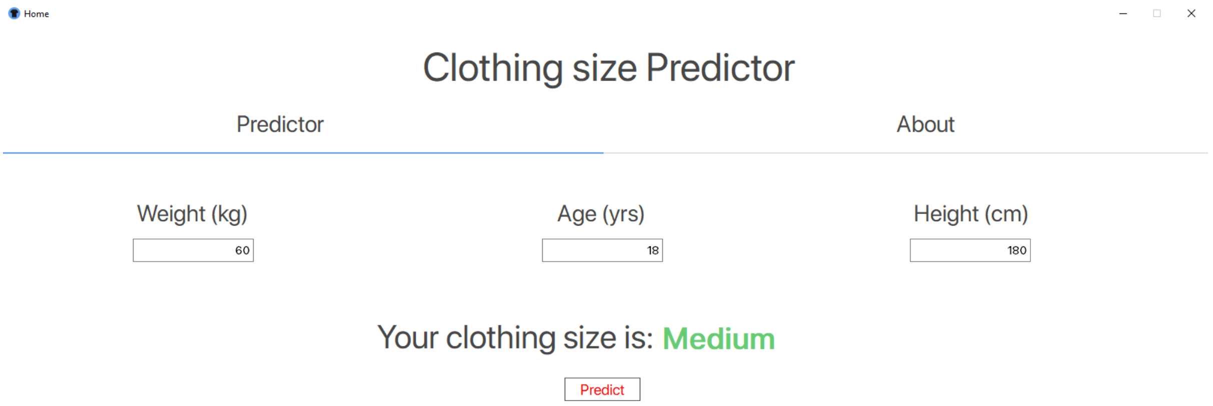


Clothing size Predictor

Made by Juan David Pisco Jaimes

JuanDavidPiscoJaimes

juan-david-pisco-jaimes



Clothing size Predictor

Weight (kg): 60

Age (yrs): 18

Height (cm): 180

Your clothing size is: Medium