Predicting Pictionary Sketches

Cecilia Li, Malo Hamon and Chelsea Le
Team Human Learner

Kaggle result

We have considered a range of models including:

- Random Forest (80% accuracy)
- XGBoost (???% accuracy)
- **Neural Network (92% accuracy)**

Approach

- Assign one model to each member to work on
- Tune hyperparamters, submit predictions, record accuracy
- Improve on model's performance to its maximum potential
- Summarise each model's overall performance
- Pick one model with highest accuracy

Model overview

Our best model was a Convolutional Neural Network model that was then further improved through bagging and boosting.

Convolutional Neural Network - CNN

- CNN is a type of Neural Network that specialises in **image recognition**.

- Instead of applying a simple pixel to pixel comparision, it applies a series of learned filters to create **feature maps, e.g. lines and shapes**.

Convolutional Neural Network - CNN

📊 There are **3 main types of layers in a CNN**:

. **Convolution layer**: allows certain features of the image to be recognised such as lines and shapes

. **Pooling layer**: summarises the presence of a feature in each small areas of the feature map and reduce dimension to limit memory and improve speed of the model.

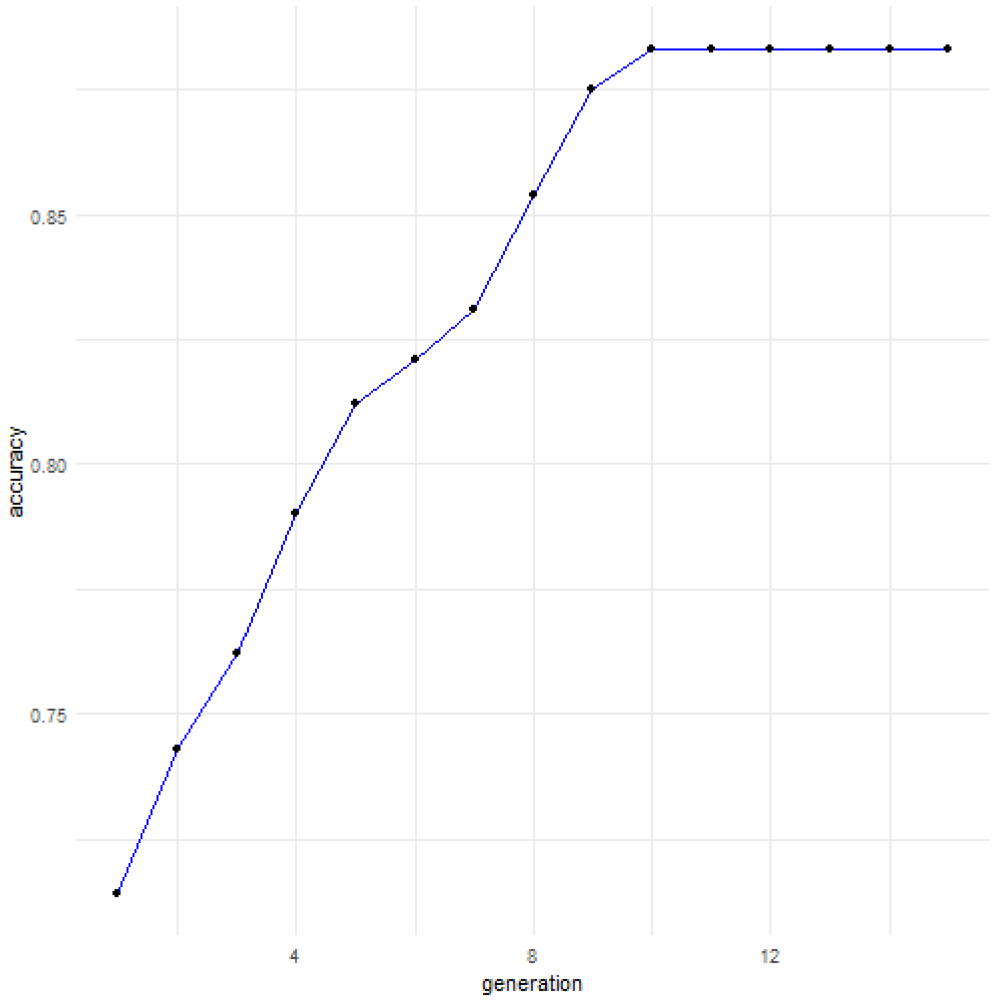
. **Fully connected layer**: is connected after a series of convolution and pooling layer to support the image classification.

Genetic algorithm

- Was used to determine the **optimal configuration for the hyperparameters of our dense layers**, i.e. found the optimal combination of:
  - the loss function,
  - optimiser,
  - the number of layers,
  - the number of nodes and activation function of each layer.

- The genetic algorithm runs through the selection and scoring of models for a **number of iterations, i.e. generations**.

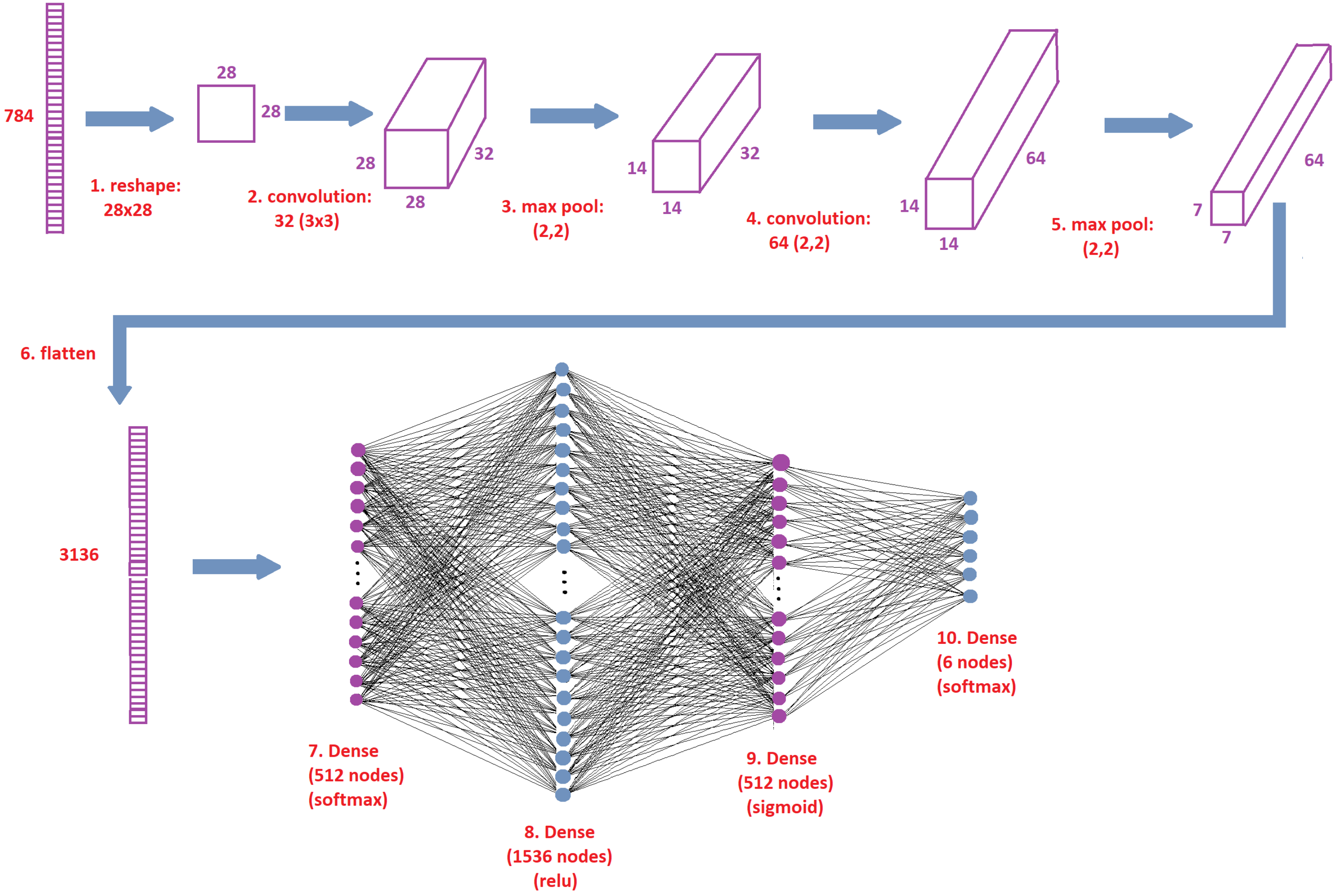
Generations vs. Accuracy:



- After 9 generations, the models stop getting better.

- This is because the population were almost all identical at this point because an optimal model had been discovered.

Full architecture of our model



Ensemble methods

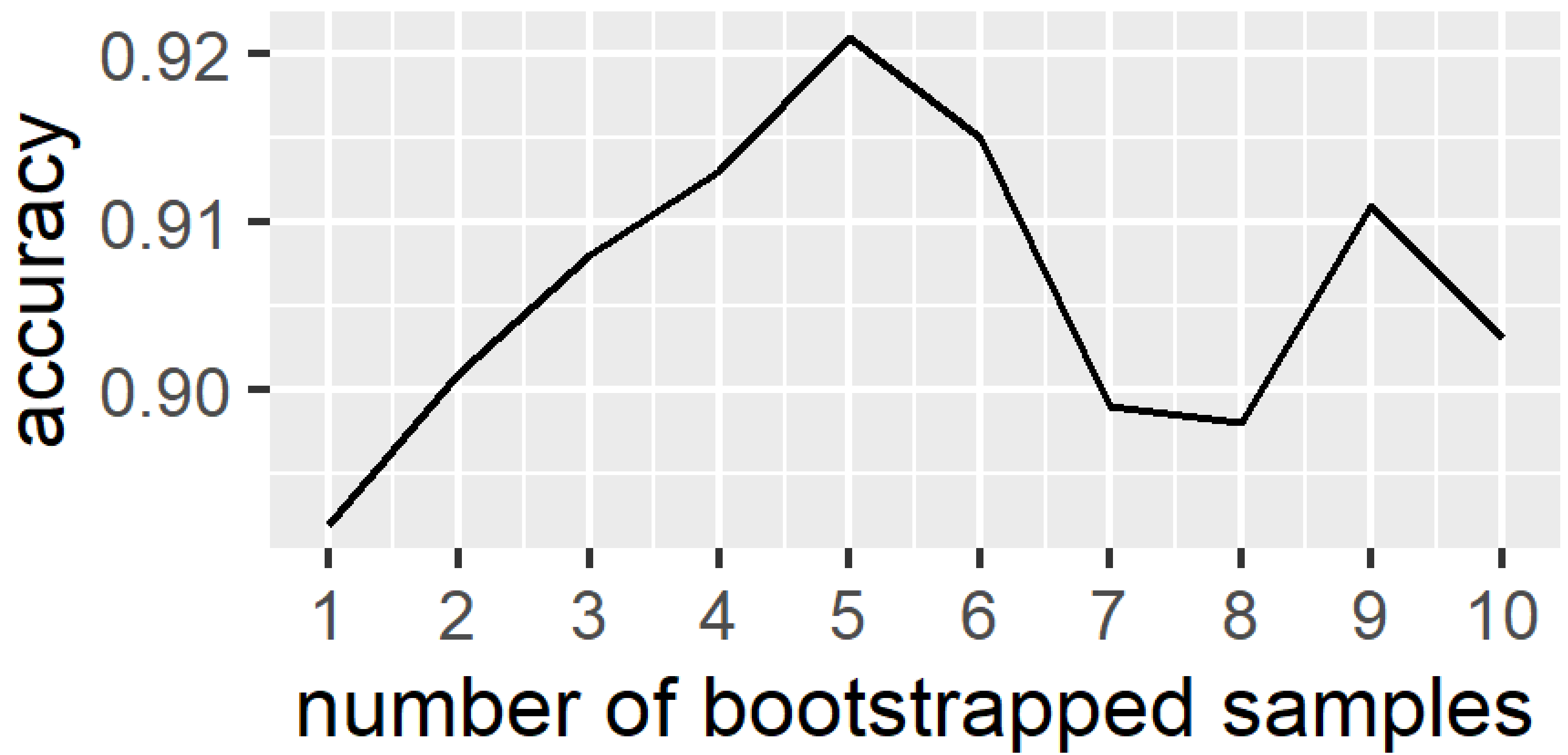
Bagging

- Train mutiple models on different boostrapped samples of the training dataset
- Calculate the average value for the reponses from each model to obtain an output
- Improvement: reduce bias

Boosting:

- Create and train a model on a boostrapped sample of the training set.
- Calculate model's accuracy with test set.
- Determine hard-to-predict observations and add them into the training set twice.
- Repeat for multiple iterations and average out the results of the first 5 boosted models.

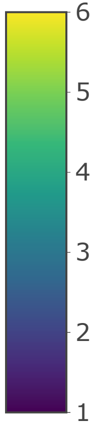
Accuracy of the ensemble method



Confusion table

Boosted + Bagged		Predicted								
		banana	boomerang	cactus	crab	flip flops	kangaroo	Total	sensitivity	specificity
Actual	banana	120	2	1	2	2	2	129	93%	99%
	boomerang	5	120	2	4	1	3	135	89%	98%
	cactus	0	0	128	3	4	3	138	93%	99%
	crab	0	1	4	125	2	2	134	93%	99%
	flip flops	1	2	1	3	120	3	130	92%	99%
	kangaroo	2	1	4	2	2	127	138	92%	98%
Accuracy	0.921							804		

Principal Component

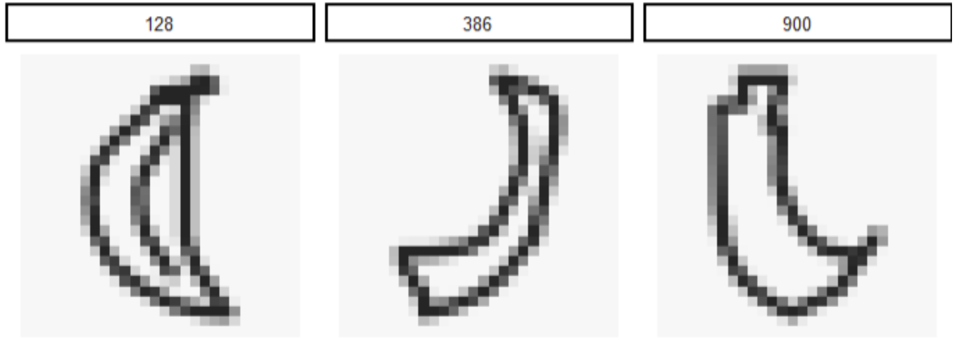


Interesting observations

The two most often confused classes are **bananas and boomerangs**, which were constantly misclassified by both RF and XGB

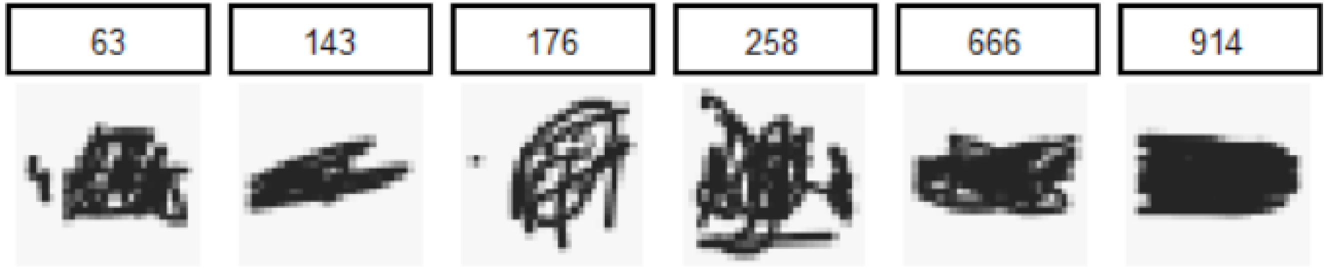
CNN, however, had little difficulty

Some examples where only CNN correctly recognised to be bananas, whereas RF and XGB labeled as boomerangs:

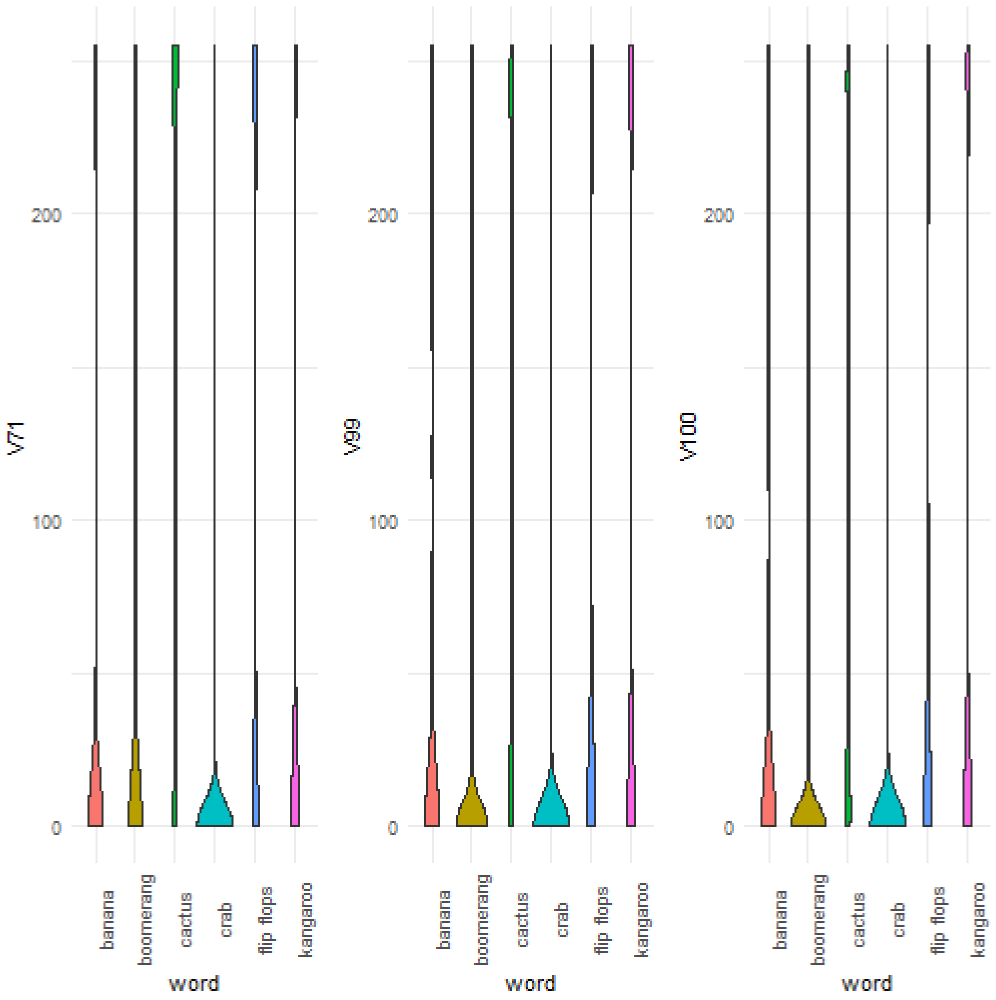


Interesting observations: scribbles

- There were some observations where it looks like the player just gave up on the game and decided to make **random scribbles**.
- These sketches are indecipherable even with human eyes and all three models could not decide on a unanimous answer.



Important variables



Acknowledgements

Slides produced using Rmarkdown with xaringan styling.

Questions?

Author: Cecilia Li, Malo Hamon and Chelsea Le