cs577 Assignment 3

Jorge Gonzalez Lopez
A20474413
Department of Computer Science
Illinois Institute of Technology
March 19, 2021

Abstract

Two different datasets, one for multi-class classification and another one for single output regression have been selected in order to test different loss functions, optimizers, and regularization techniques to determine how do they affect the neural network results based on the data selected.

Problem statement:

Two Neural network are created using Keras for multi-class classification and regression to test the following specifications of the network:

- Loss function: For the regression task the loss functions that are going to be tested are MAE, MSE, Huber Loss and Log-cosh. For the classification task the functions are categorical cross-entropy, Kullback-Leibler, Hinge and Squared Hinge.
- Optimizer: The optimizers that are going to be analyzed for both tasks are SGD, RMSprop, Adagrad, Adam, Adadelta, Adamax and Nadam.
- Regularization technique: The regularization techniques that are going to be implemented and compared are L1, L2, Dropout, Batch normalization and ensemble:
    - L1 and L2 are going to be added in all the layers with a regularization rate of 0.01.
    - A Dropout layer is going to be implemented after every hidden layer with a dropout rate of 25%.
    - A Batch Normalization layer is also going to be implemented after every hidden layer.
    - The ensemble classifier is going to be the mean of the L1, L2 and Batch Normalization classifiers.

Proposed Solution:

The datasets selected for both tasks are the following:

1. Wine Quality Data Set:

   This dataset consists of 11 features that measure some important features (acidity, density, pH, alcohol…) of 4898 different wines to try to get the wine's quality between 0 and 10. This problem can be solved either with a regression or a classification model. Hence, in this case, a regression approach has been followed.

At the same time, there are some features with really low values and other with much greater values. For example, the feature “chlorides” has values around 0.04 and “total sulfur dioxide” of around 110. Therefore, a normalization of each feature has been carried out by subtracting their means and divide by their standard deviation.

Finally, the data is split into inputs (X) and the target (y), shuffled randomly, and divided in three different datasets: train, validation, and test (with a ratio 70:15:15).

2. Semeion Handwritten Digit Data set:

   This dataset consists of 1593 handwritten digits (from 0 to 9) from around 80 people stretched in a rectangular box 16x16 in a gray scale of 256 values. However, the values of the pixels have been already normalized by setting to 0 every pixel whose value was under the value 127 of the grey scale (127 included) and setting to 1 each pixel whose original value in the grey scale was over 127.

   At the same time, the target is a vector of length 10 corresponding to the one-hot encoding of the possible handwritten digits.

   Therefore, as with the last dataset, the data has been shuffled and split into three datasets: train, validation, and test (with a ratio 70:15:15).

Once the data has been correctly loaded, a simple neural network has been created and tuned in order to get in both tasks a low training error (even if there is some overfitting) and a smooth convergence. After several trials, the model chosen is the following one:

- First Dense hidden layer with 32 neurons and ‘ReLu’ activation.
- Second Dense hidden layer with 64 neurons and ‘ReLu’ activation.
- Third Dense hidden layer with 128 neurons and ‘ReLu’ activation.
- Forth Dense hidden layer with 256 neurons and ‘ReLu’ activation.
- Output Dense layer:
  - With a single neuron and no activation for the Regression task.
  - With 10 neurons and ‘softmax’ activation for the Classification task.

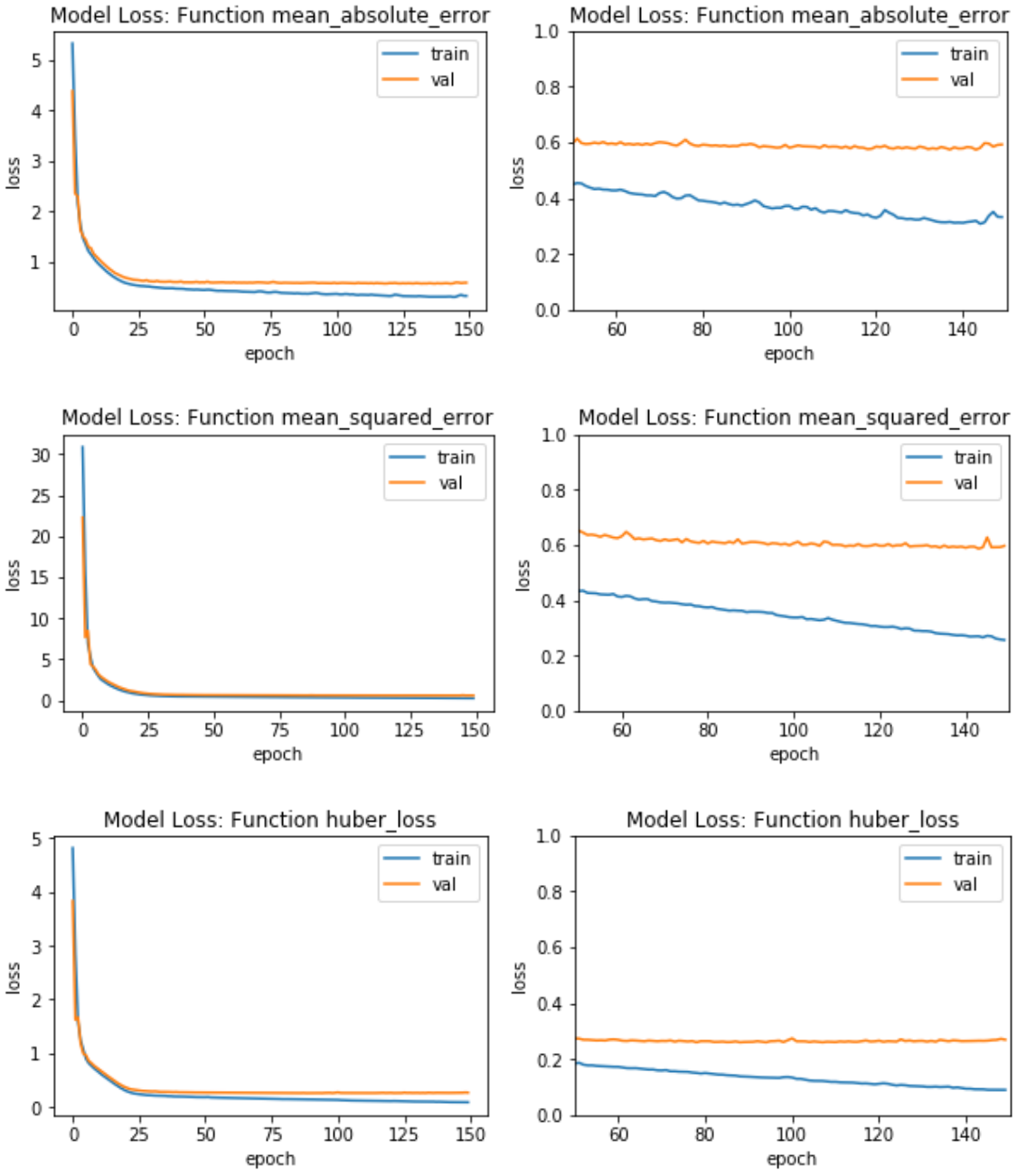
Now, the following approach is going to be used. First, the previous model, without any regularization is going to be trained with the Adam optimizer and with all the different losses mentioned earlier. Then, the results are going to be analyzed and the loss function that outputs the best results is going to be chosen. With this loss function, again with no regularization, the model is going be trained with all the optimizers. Finally, once the best loss function and optimizer have been chosen, all the regularization techniques are going to be tried out to choose the one that outputs the best results.

Results and Discussion:

1. Wine Quality Data set:

To check the results of each loss function, optimizer and regularization technique, the evolution of the values of the training and validation losses are going to be plotted. The graphs at the left show all the values and the graphs at the right just show the final values of both losses.

The results with different losses are the following:



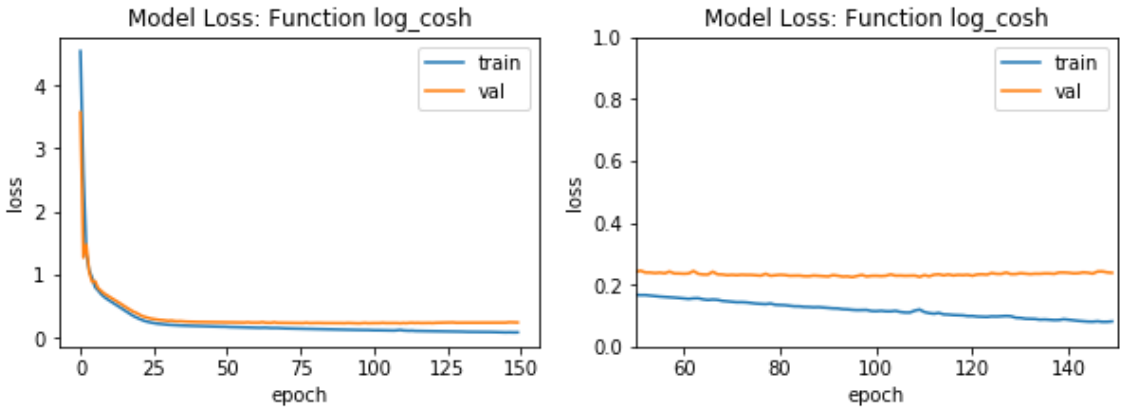
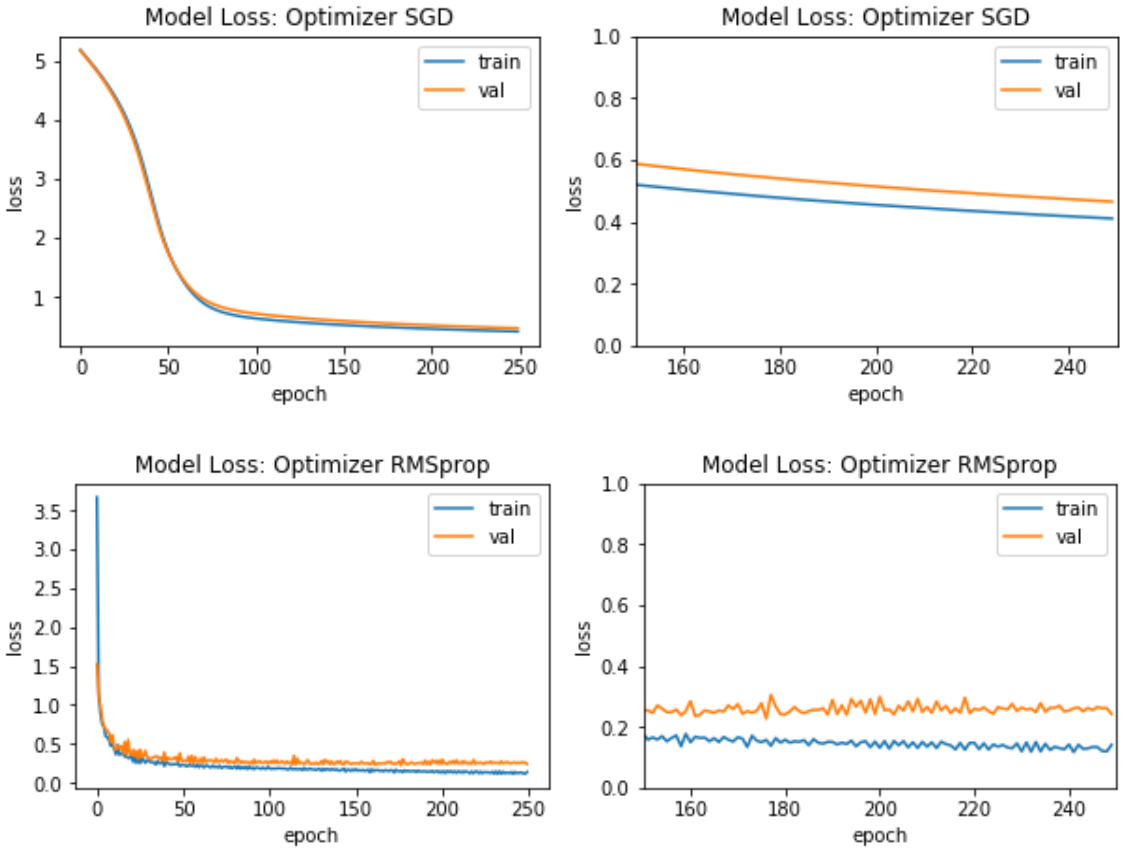
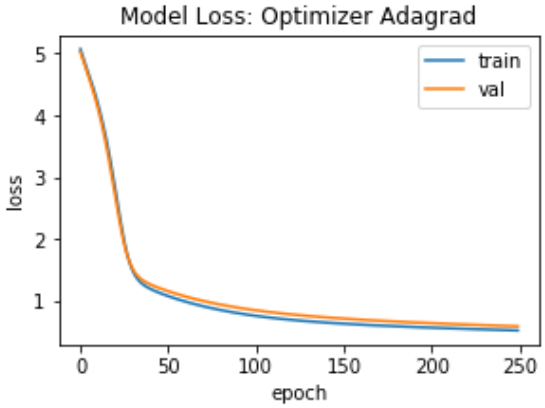


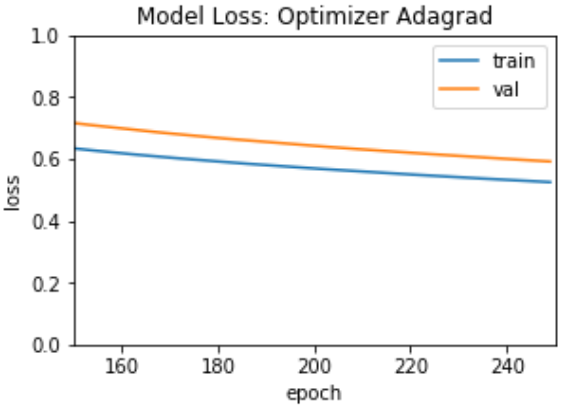
Figure 1: Performance of the loss functions over the regression task.

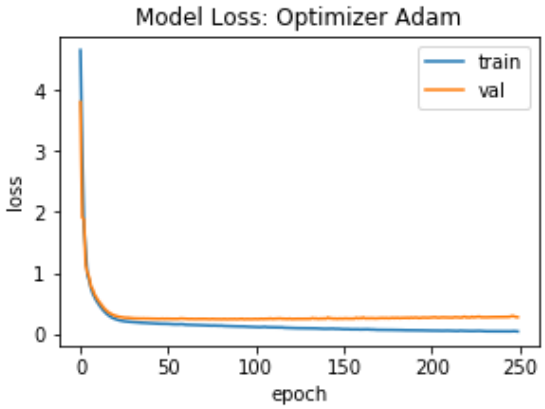
From the previous graphs, the best losses are the ones that draw out the lower final values for both, the training and validation errors. Those losses are the Huber loss and the log-cosh, of which the log-cosh is the one selected for the next tests.

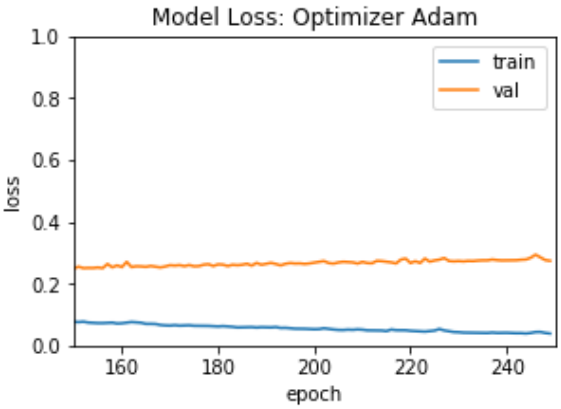
Now, different optimizers are going to be tested. The learning rate of all of them has been fixed to 1e-3 except for Adadelta, in which it has been fixed to 1e-2 so it could converge in the same number of epochs as the rest of the optimizers. The results are shown in the following graphs:

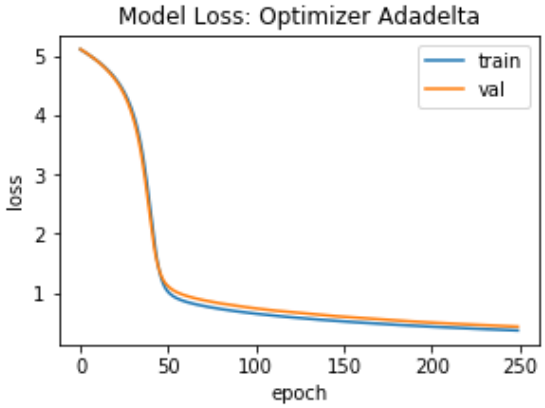


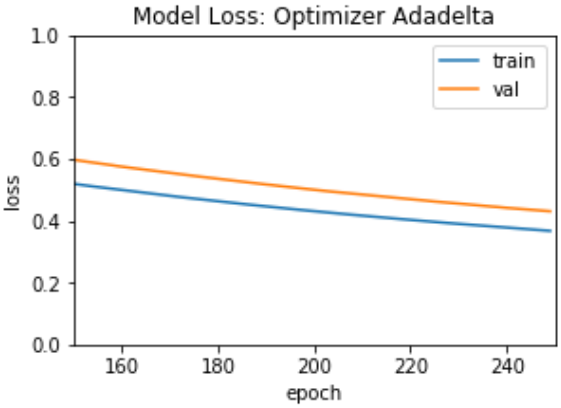

Model Loss: Optimizer Adagrad
train
val
loss
epoch

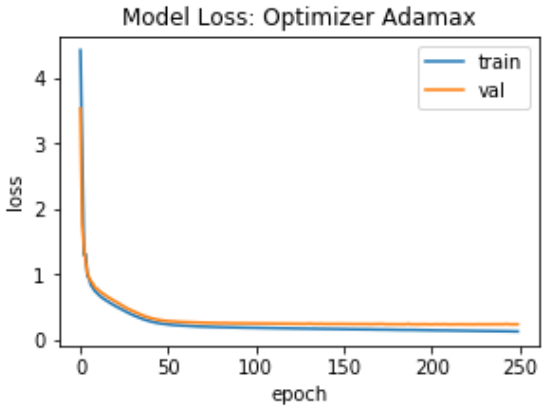

Model Loss: Optimizer Adagrad
train
val
loss
epoch

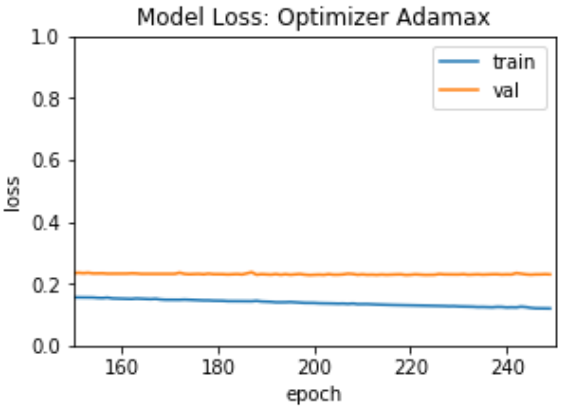

Model Loss: Optimizer Adam
train
val
loss
epoch


Model Loss: Optimizer Adam
train
val
loss
epoch


Model Loss: Optimizer Adadelta
train
val
loss
epoch


Model Loss: Optimizer Adadelta
train
val
loss
epoch


Model Loss: Optimizer Adamax
train
val
loss
epoch


Model Loss: Optimizer Adamax
train
val
loss
epoch

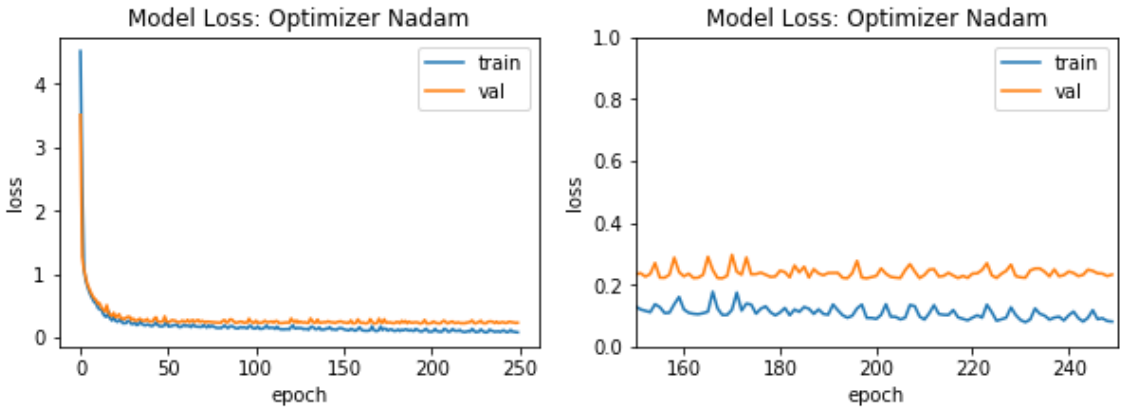
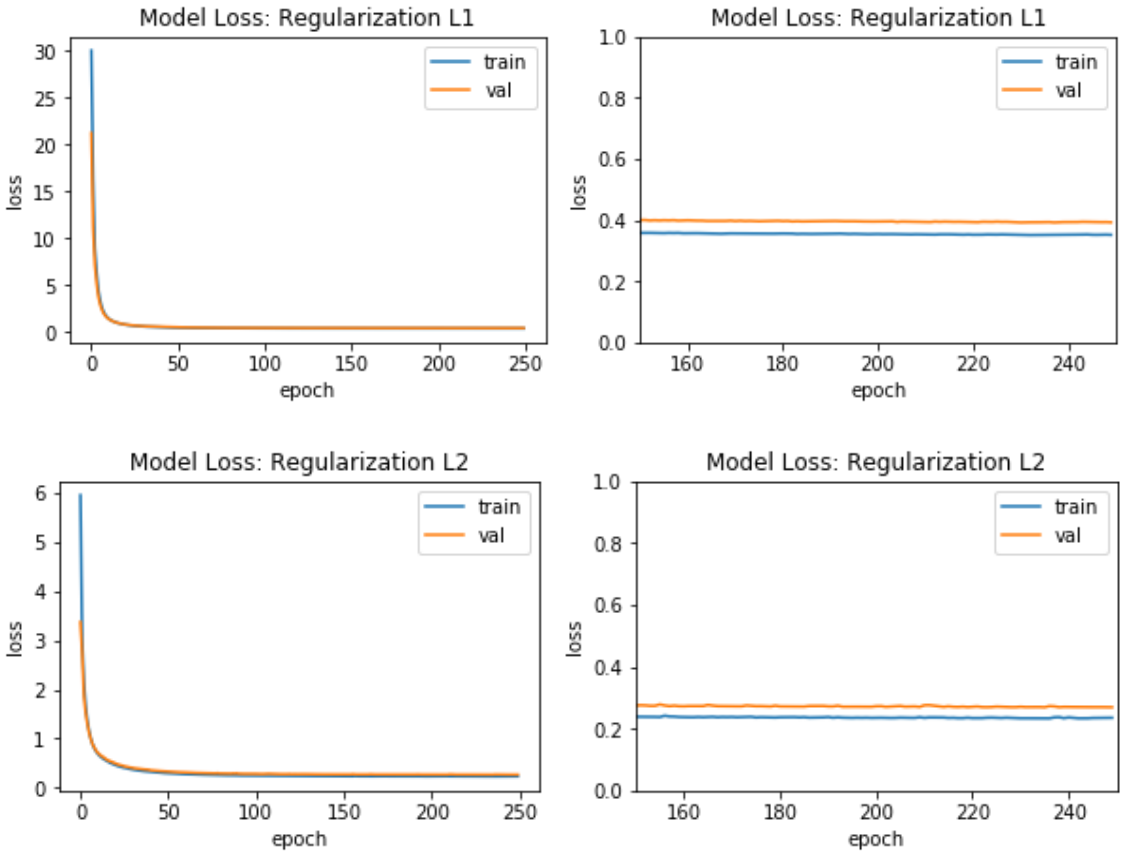


Figure 2: Performance of the optimizers over the regression task.

Again, the best optimizers are the ones that draw out the lower final values for both, the training and validation errors, and show a smooth convergence onto those values. The best optimizers in this case are Adam and Adamax, of which the Adam is the one selected for the next tests as it has the lower training error.

Finally, different regularization techniques are going to be tested in order to see which of them improve the results the most and make the model generalize better to new data:



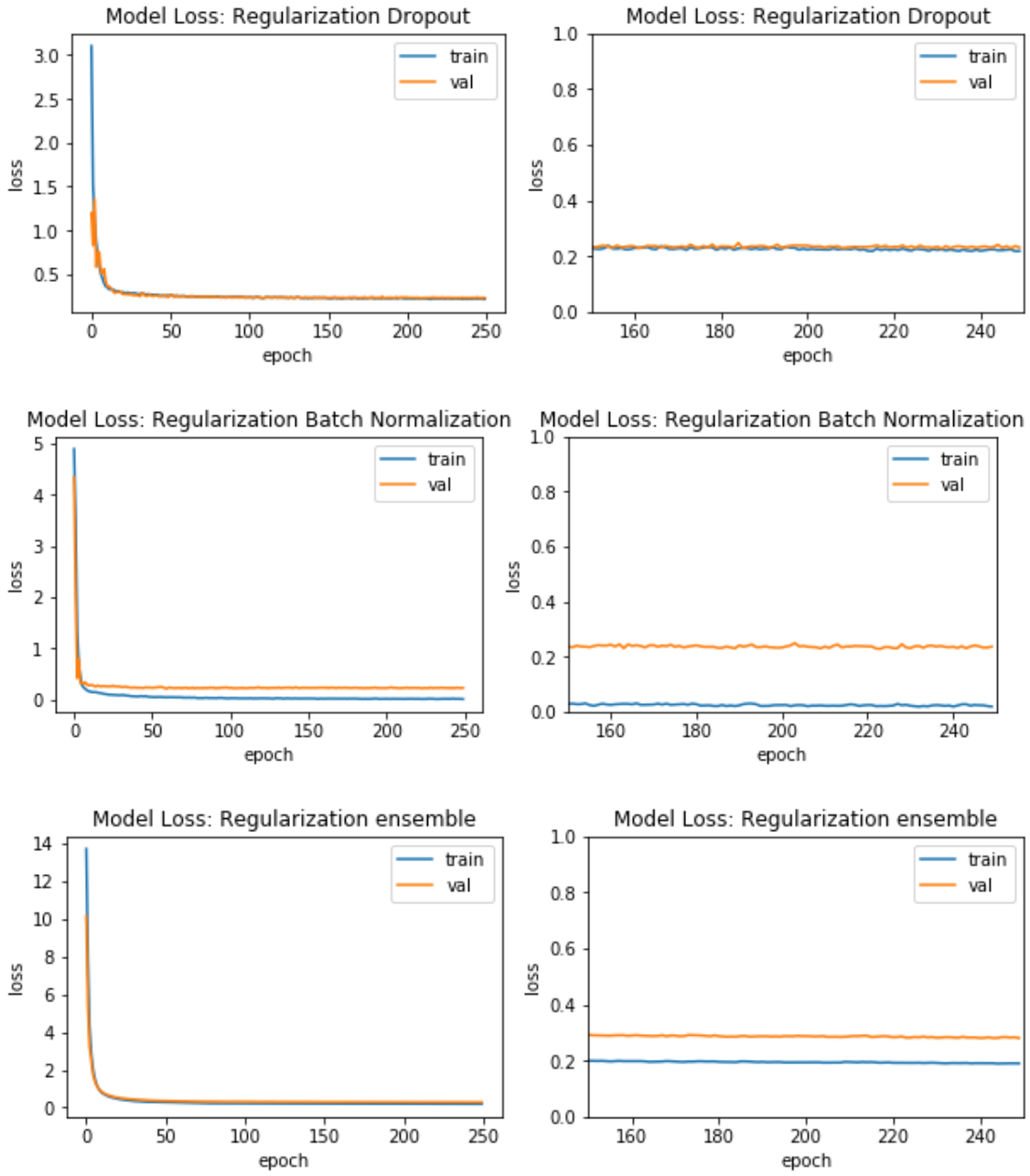


Figure 3: Performance of the regularization techniques over the regression task.

In this case, the best results are the ones that achieve a low error but similar in both datasets (training and validation) as that implies that the model performs with new data as it does with the data with which it has been trained. Therefore, the regularization technique chosen is Dropout.

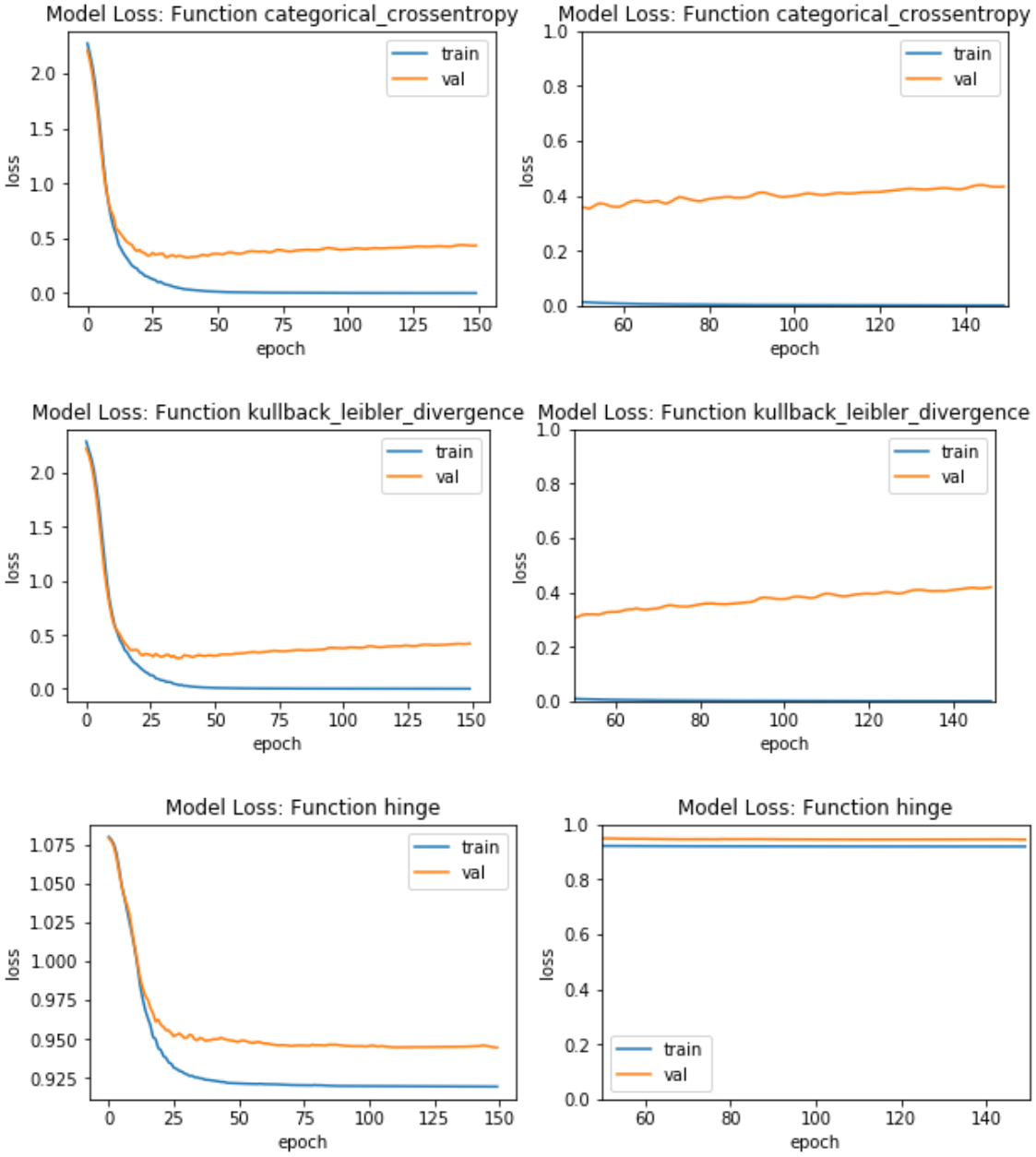
Finally, with the log-cosh loss function, the Adam optimizer and the Dropout regularization, the results over the test dataset are the following:

```
loss: 0.1895
```

2. Semeion Handwritten Digit Data Set:

As before, to check the results of each loss function, optimizer and regularization technique, the evolution of the values of the training and validation losses are going to be plotted. The graphs at the left show all the values and the graphs at the right just show the final values of both losses.

The results with different losses are the following:



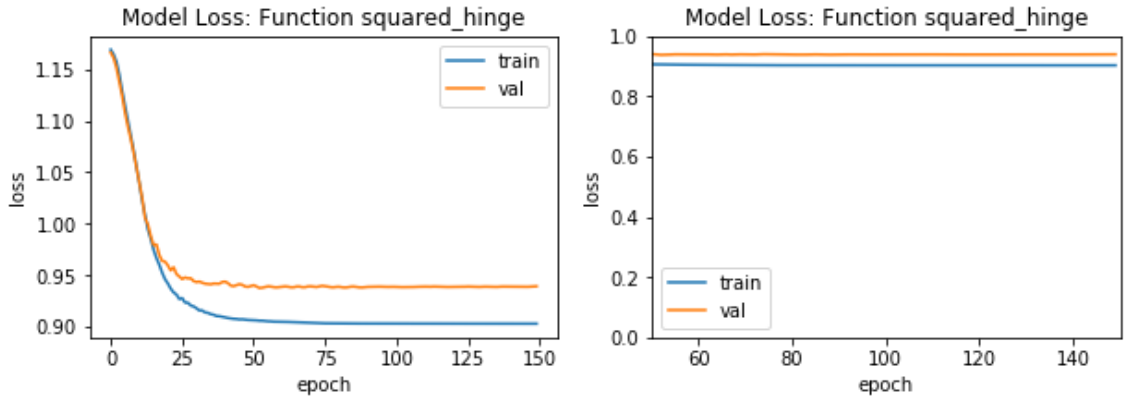
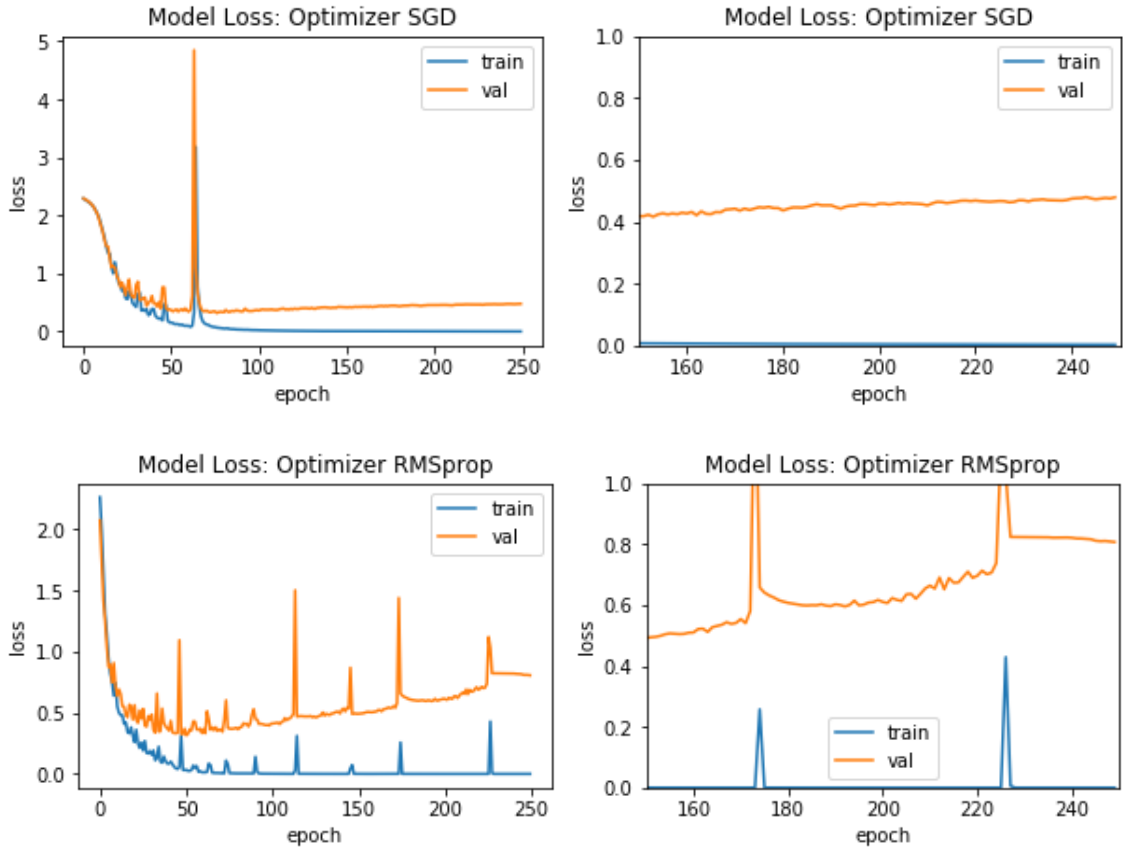
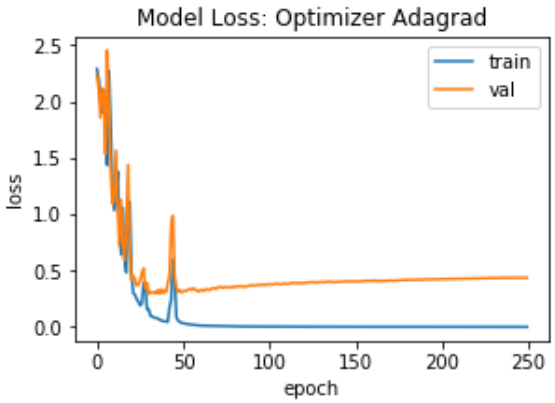


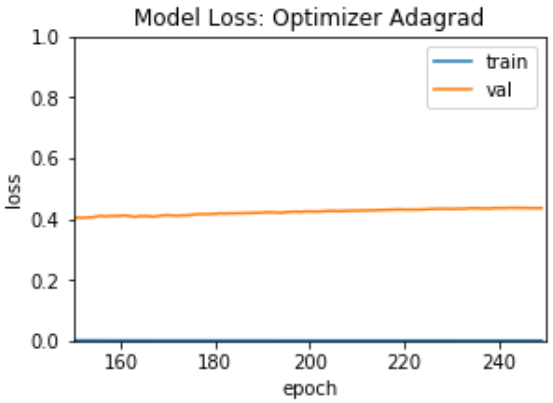
Figure 4: Performance of the loss functions over the classification task.

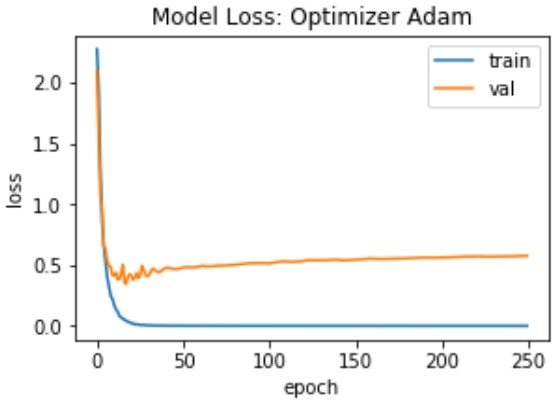
In this case, the best losses are categorical cross entropy and Kullback Leibler, of which the Kullback Leibler is the one selected for the next tests.

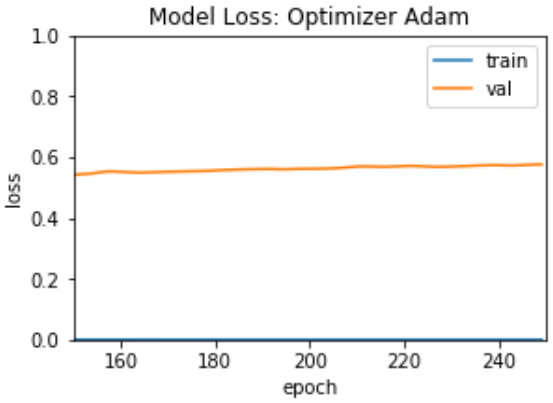
Now, different optimizers are going to be tested. To make them converge in the same number of epochs, the learning rate of SGD and Adagrad has been tuned to 0.1, Adadelta's one to 0.5 and the rest of them to 1e-3. The results are shown in the following graphs:

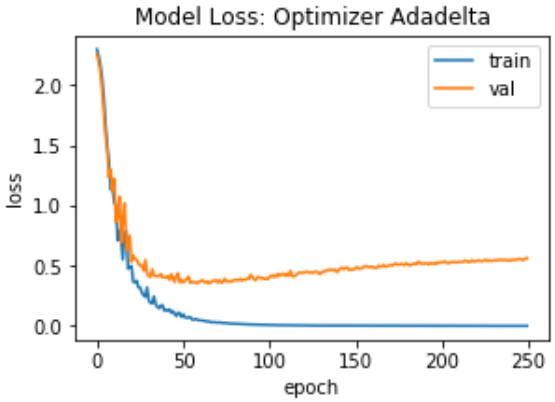



Model Loss: Optimizer Adagrad
train
val
loss
epoch

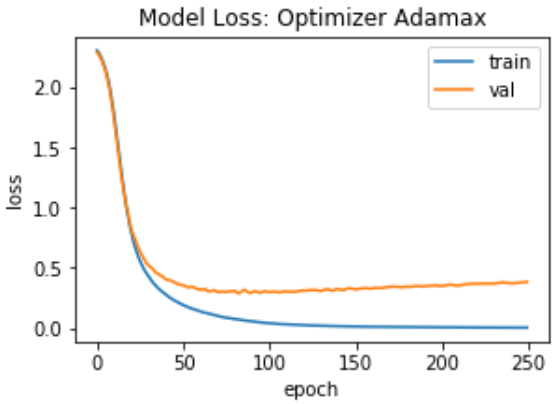

Model Loss: Optimizer Adagrad
train
val
loss
epoch


Model Loss: Optimizer Adam
train
val
loss
epoch


Model Loss: Optimizer Adam
train
val
loss
epoch


Model Loss: Optimizer Adadelta
train
val
loss
epoch


Model Loss: Optimizer Adadelta
train
val
loss
epoch


Model Loss: Optimizer Adamax
train
val
loss
epoch


Model Loss: Optimizer Adamax
train
val
loss
epoch

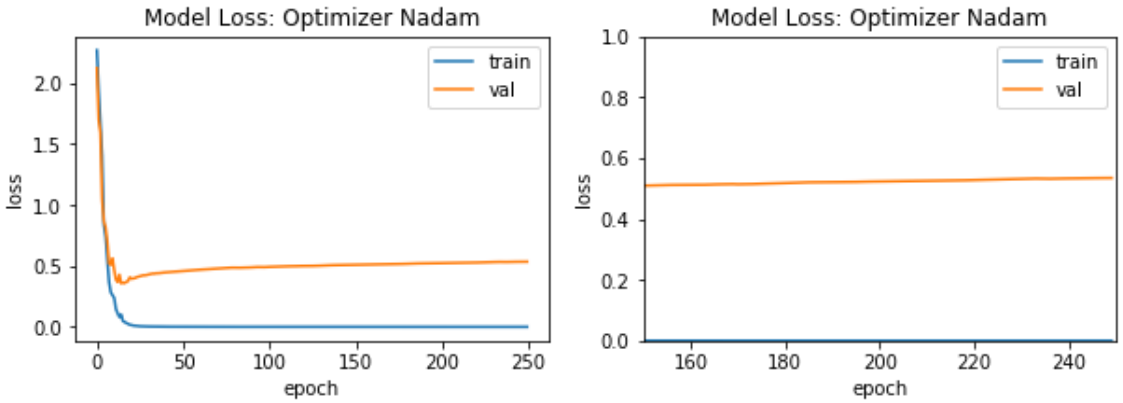
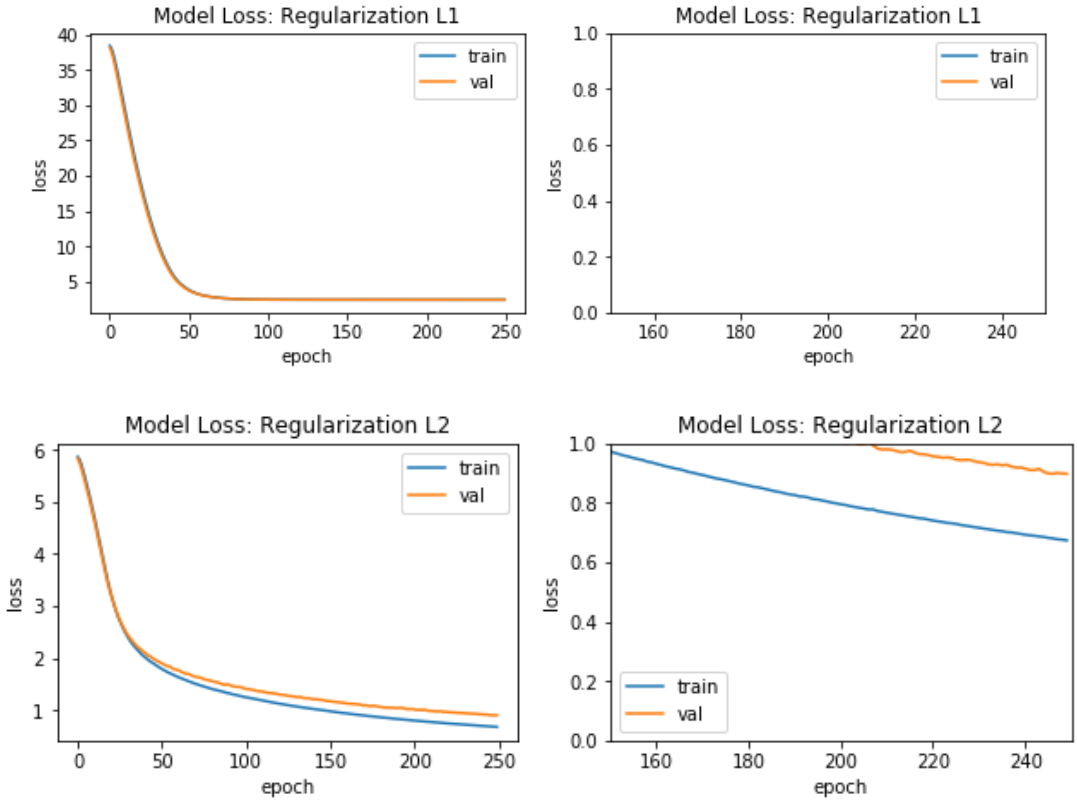


Figure 5: Performance of the optimizers over the classification task.

Again, the best optimizers are the ones that draw out the lower final values for both, the training and validation errors, and show a smooth convergence onto those values. The best optimizers in this case are Adagrad and Adamax, of which the Adamax is the one selected for the next tests as it has the lower training error.

Finally, different regularization techniques are going to be tested in order to see which of them improve the results the most and make the model generalize better to new data:



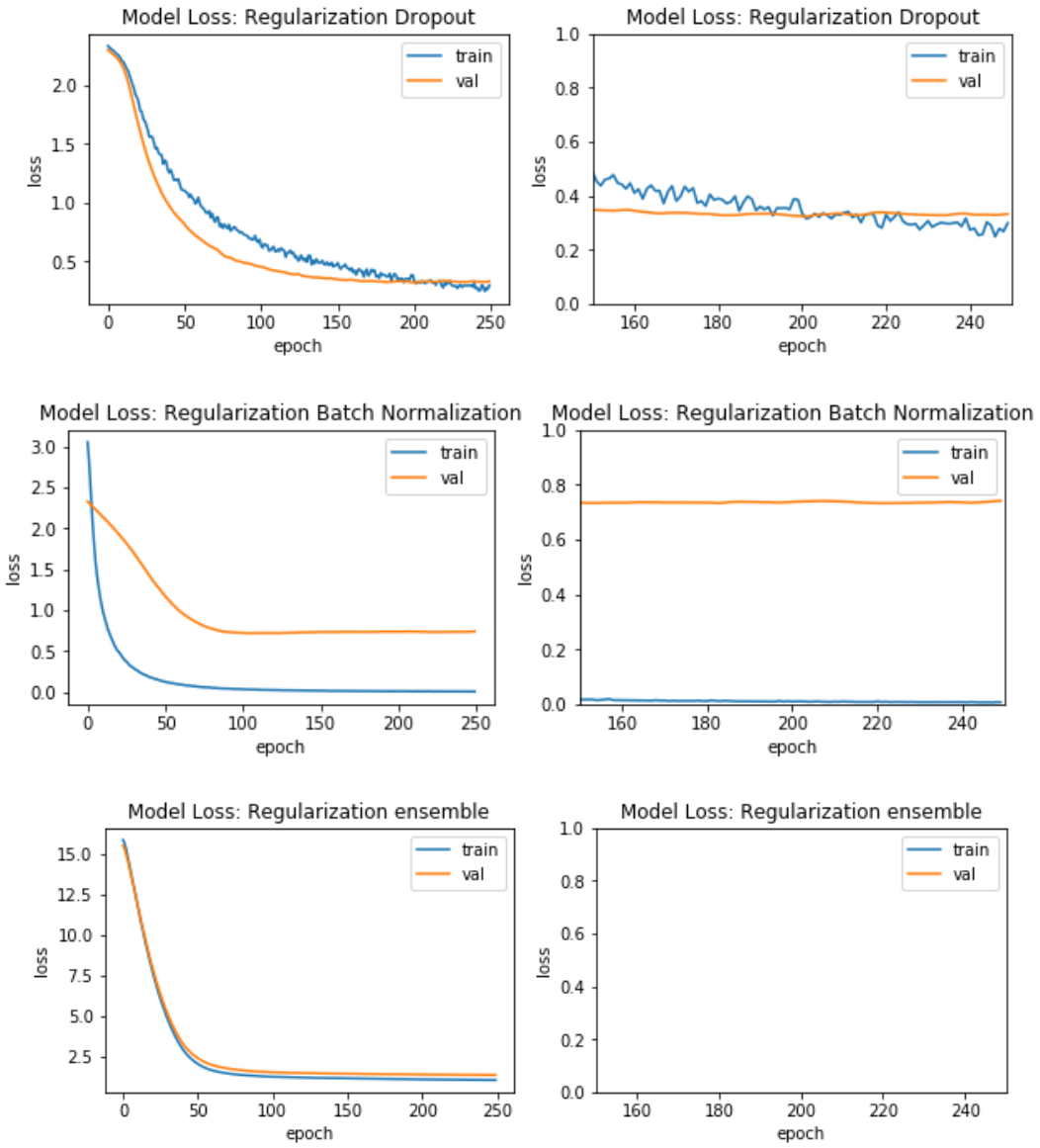
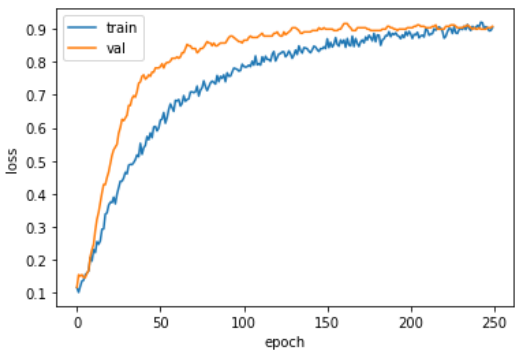


Figure 6: Performance of the regularization techniques over the classification task.

The regularization technique chosen is Dropout. And with the Kullback Leibler loss function, the Adamax optimizer and the Dropout regularization, the accuracy over the training and validation sets and the results over the test dataset are the following:




```
Results over Test Dataset:
loss: 0.3519 - accuracy: 0.8875
```

Figure 7: Evolution of accuracy over the training and validation sets and final results over test set.