1 Hyperparameters optimization

1.1 Linear regression

There are no hyperparameters in closed form linear regression. The MSE on the test data is 558.059. We will discuss this result in the dedicated section.

1.2 Ridge regression

For ridge regression, the only hyperparameter to be optimized with the help of cross-validation is the regularization parameter lambda.

Figure 3 is the validation loss based on lambda. We can see clearly the best value for lambda is in between 170-175.

This demonstrates the overfitting/underfitting dilemma on the training data: if the regularization factor is too low the ridge regression overfits the training data which results in a higher loss when tested on data that has never been seen before (here the validation set). On the contrary, when the lambda is too big, it puts too much restriction on the model which prevent it from learning properly.

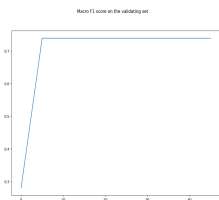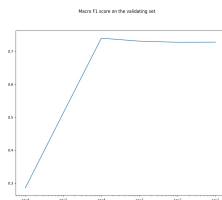## 1.3 Logistic Regression

Figure 1

Figure 2

For logistic regression, we can optimize the learning-rate as well as the number of iterations.

Figure 1 shows the F1 Score Based on the number of iterations with fixed learning rate 1e-4. It reaches a minimum under 10 iterations and is stable.

Figure 2 shows the F1 score based on the learning rate used in gradient descent. There is a maximum value close to 1e-4. The cross validation was done with enough iterations in order to not penalize a lower learning-rate since they would need more time to converge.
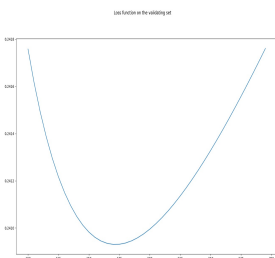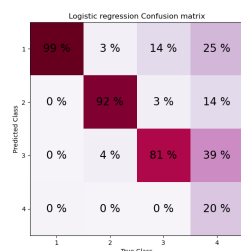

Figure 3

Figure 4

2 Results and discussion

linear regression	ridge regression	logistic regression
		lr = 1e−4
	$\lambda = 170$	max-iterations = 10
		Accuracy = 78.94 %
MSE = 558.059	MSE = 0.367	Macro F1 = 0.7100

2.1 MSE on ridge/linear regression

There is a big gap in loss between ridge regression with $\lambda = 0$ and $\lambda = 170$. This can be explained by the very high dimensionality of H36m data compared to the number of samples. Without any regularization the model will poorly generalize which results in a very high loss on the test data.

2.2 Logistic regression

We got satisfying results on logistic regression with an accuracy close to 80% and F1 score of 0.71. However, if we look closer at the confusion matrix in Figure 4, we can see that it struggles to classify class 4 correctly. We think this poor performance is in part due to the lower amount of class 4 data compared to the other classes. (see Figure 5). Performances for classes 1 and 2 on the other hand are excellent with an accuracy close to 100%.

2.3 Cross-validation limitations

class	1	2	3	4
n-labels	809	770	550	460

class	1	2	3	4
n-labels	131	222	163	102

Figure 5: training labels

Figure 6: test labels

Cross-validation is a fantastic tool, but in practice, it can have a few limitations. Indeed, after finding optimal parameters with this method, we tried tweaking manually the parameters on the test data to see how good of an optimization cross-validation did. To our surprise, there were some noticeable but not major differences.

Since cross-validation only use training data to validate itself, if there's a difference in features distribution between the test data and the training data, the optimization won't be perfectly accurate. If we look at figure 5 and figure 6 we can see that the distribution between classes is quite different. This may be one of the reason our model performed slightly worse than expected on the test data where the different labels are more evenly distributed.