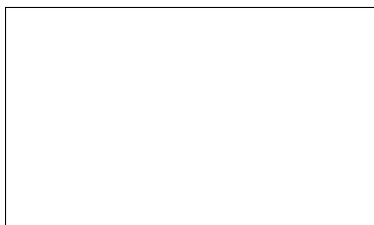


## Graphical Abstract



### **Application of Machine Learning to the discovery of new physics**

Óliver Partida, Pere Masjuan



## Highlights

### **Application of Machine Learning to the discovery of new physics**

Óliver Partida,Pere Masjuan

- Research highlights item 1
- Research highlights item 2
- Research highlights item 3

# Application of Machine Learning to the discovery of new physics

Óliver Partida<sup>a,c,\*,1</sup> (Researcher), Pere Masjuan<sup>b,d</sup>

<sup>a</sup>Elsevier B.V., Radarweg 29, 1043 NX Amsterdam, The Netherlands

## ARTICLE INFO

### Keywords:

standard model  
B-physics  
neural networks  
conditional adversarial networks

## ABSTRACT

This template helps you to create a properly formatted L<sup>A</sup>T<sub>E</sub>X manuscript.

`\beginabstract ... \endabstract` and `\begin{keyword} ... \end{keyword}` which contain the abstract and keywords respectively.

Each keyword shall be separated by a `\sep` command.

## 1. Introduction

Flavour Changing Neutral Current (FCNC) are interactions that change the flavor of a fermion without altering its electric charge. FCNC processes are forbidden at tree level in the Standard Model and highly suppressed at higher orders by the GIM mechanism. The GIM mechanism required the existence of a fourth quark in order to explain the suppression in loop diagrams of FCNC.

This makes FCNC one of the key processes to search for new physics. Any small deviations from the Standard Model expectations could have a big impact. Over the last few years, many observables related to the FCNC transitions  $b \rightarrow sl^+l^-$  have exhibited deviations from SM expectations. Due to their suppression within the SM, these transitions are well known to have a high sensitivity to potential NP contributions. These anomalies can be classified in two sets:  $b \rightarrow s\mu\mu$  related to observables testing only muonic transitions, called Lepton Flavor Dependent (LFD), and Lepton-Flavor Universality Violating (LFUV) anomalies that correspond to deviations in observables comparing muonic and electronic transitions. In 2013, using the  $1 \text{ fb}^{-1}$  dataset, the LHCb experiment measured the basis of optimized observables for  $B \rightarrow K^*\mu\mu$ , observing the so-called  $P'_5$  anomaly, i.e., a sizable  $3.7\sigma$  discrepancy between the measurement and the SM prediction in one bin for the angular observable  $P'_5$  (Figure 1). A new discrepancy in the ratio  $R_K^* = Br(B \rightarrow K^*\mu\mu)/Br(B \rightarrow K^*ee)$  (Figure 2) was also observed by LHCb, hinting at the violation of Lepton Flavor Universality (LFU) and suggesting that deviations from the SM are predominantly present in  $B \rightarrow K\mu^+\mu^-$  transitions but not in  $B \rightarrow Ke^+e^-$  ones. In order to evaluate the significance and coherence of these deviations, a global model-independent fit is the most efficient tool to determine if they contain patterns explained by New Physics (NP). The starting point is an effective Hamiltonian in which heavy degrees of freedom (the top quark, the  $W$  and  $Z$  bosons, the Higgs and any heavy new particle) are integrated out in short-distance Wilson coefficients  $C_i$ , leaving only a set of operators  $\mathcal{O}_i$  describing the physics at long distances:

scribing the physics at long distances:

$$\mathcal{H}_{eff} = -\frac{4G_F}{\sqrt{2}} V_{tb} V_{ts}^* \sum_i C_i \mathcal{O}_i.$$

In the SM, the Hamiltonian contains 10 main operators with specific chiralities due to the V - A structure of the weak interactions. In presence of NP, additional operators may become of importance. Current analysis of anomalies in flavour physics are based on a linear regression of a  $\chi^2$  function. After taking into account correlation between theoretical predictions and experimental observables the  $\chi^2$  is built and minimized. For each measured observable, we have a theory prediction based on the Standard Model of Particle Physics (SM). With 180 observables, the  $\chi^2$  value of the SM reaches 225 points [1], which corresponds to a p-value of 1.4%. This indicates the SM to be very far to explain experimental measurements. The strategy then has been to include on top of the SM, new operators in the effective Hamiltonian to be able to account for such experimental discrepancies. In [1] including measurement updates ( $R_K, R_K^*$  and  $B(B_s \rightarrow \mu^+\mu^-)$ ) a global model-independent analysis is performed yielding very similar results to the ones previously found in Refs. [2],[3] for the various NP scenarios of interest.

For the processes considered here, we focus our attention on the semileptonic operators  $\mathcal{O}_9$  and  $\mathcal{O}_{10}$ :


$$\mathcal{O}_9 = \frac{e}{16\pi^2} m_b (\bar{s} \gamma_\mu P_L b) (\bar{\ell} \gamma^\mu \ell),$$

$$\mathcal{O}_{10} = \frac{e}{16\pi^2} m_b (\bar{s} \gamma_\mu P_L b) (\bar{\ell} \gamma^\mu \gamma_5 \ell).$$

## 2. Machine learning techniques: Neural Networks And Generative Adversarial Networks

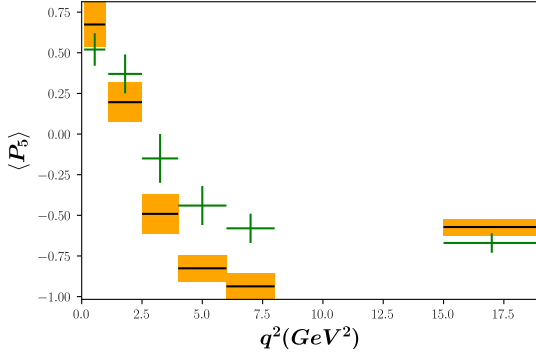
### 2.1. Generative vs. Discriminative models

Generative and discriminative models are the two main types of models used to solve machine learning classification problems. In classification tasks we are usually interested in learning the conditional distribution  $p(C_k | \mathbf{x})$ , where  $\mathbf{x}$  is the observation and  $C_k$  is the corresponding class assigned to this observation, and then use this conditional distribution

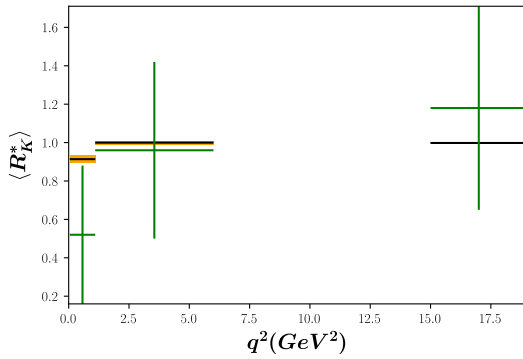
 cvr\_1@tug.org.in (Ó. Partida)

 www.cvr.cc, cvr@sayahna.org (Ó. Partida)

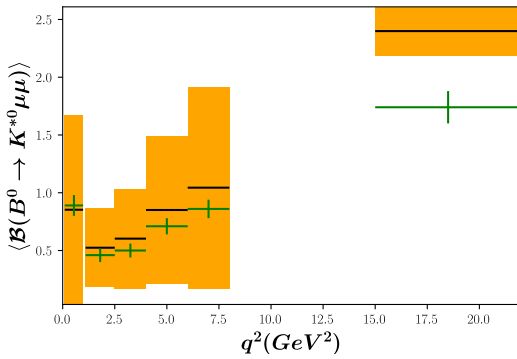
ORCID(s): 0000-0001-7511-2910 (Ó. Partida)



**Figure 1:**  $P_5 = B \rightarrow K^* \mu \mu$ . SM predictions (orange boxes) and experimental values (green crosses).



**Figure 2:**  $R_K^* = Br(B \rightarrow K^* \mu \mu) / Br(B \rightarrow K^* ee)$ . Ratios with different lepton at the final state,  $R_K$  ratios, are a clear indication of the Lepton Flavor Universality Violation something which is not expected from the theory.



**Figure 3:**  $B(B^0 \rightarrow K^{*0} \mu \mu)$ . Branching ratios.

to make class assignments. Generative models tackle this problem by first determining the class-conditional distributions  $p(\mathbf{x}|C_k)$  and the prior class probabilities  $p(C_k)$  and the using Bayes' theorem

$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{p(\mathbf{x})}$$

to find the posterior class probabilities  $p(C_k|\mathbf{x})$ . It is also possible to model the joint distribution  $p(\mathbf{x}, C_k)$  directly and normalize to obtain the posterior probabilities. Data from the input space can be generated by sampling from the learned model.

Discriminative classifiers, on the other hand, model the posterior  $p(y|x)$  directly or learn a direct map from inputs  $x$  to class labels. Discriminative models that use probability distributions to solve the classification problem are called probabilistic discriminative models. For example, in the case of two classes, the posterior probability

$$p(C_1|x) = \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x}|C_1)p(C_1) + p(\mathbf{x}|C_2)p(C_2)}$$

can be rewritten as

$$\frac{1}{1 + \exp(-a)} = \sigma(a),$$

where

$$a = \ln \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x}|C_2)p(C_2)}$$

and  $\sigma(a)$  is the logistic sigmoid function. For some specific conditional distributions  $p(\mathbf{x}|C_k)$  the argument of the sigmoid function is a linear function of the inputs  $\mathbf{x}$

$$a = \mathbf{w}^T \mathbf{x} + \mathbf{w}_0.$$

In discriminative modeling we can use maximum likelihood to directly find the parameters  $w$ .

## 2.2. Neural Networks

Usually linear models for regression and classification are based on linear combinations of fixed nonlinear basis functions

$$y(\mathbf{x}, \mathbf{w}) = f\left(\sum_{j=1}^M w_j \phi_j(\mathbf{x})\right),$$

where  $f(\cdot)$  is a nonlinear activation function in the case of classification and the identity in the case of regression. For example in polynomial regression when there is only one input variable the set of basis function  $\phi_j(\mathbf{x})$  take the form:

$$[1, x, x^2, x^3 \dots]$$

One limitation of polynomial basis functions is that they are global functions of the input variable so input space regions are not independent. This problem can be alleviated by fitting different polynomials to different regions of space leading to *spline functions*. Neural networks make basis functions depend on adjustable parameters that are *learned* along network coefficients  $w_j$  during training. In the basic neural network model with just one hidden layer the output of each basis function is the result of applying a nonlinear function  $h(\cdot)$  to a linear combination of the input variables  $x_1, \dots, x_D$ :

$$a_j = \sum_{i=1}^D w_{ji} x_i x_{j0}$$

$$z_j = h(a_j).$$

These values are again linearly combined and transformed using an appropriate function to give the final output:

$$a_k = \sum_{i=1}^D w_{kj} z_j w_{k0}$$

$$y_k = g(a_k).$$

For regression problems  $g$  is the identity so  $y_k = a_k$ . For binary classification  $g$  is the sigmoid function:

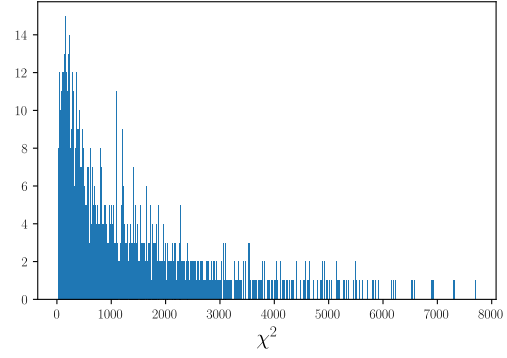
$$g(a) = \frac{1}{1 + \exp(-a)}.$$

The goal of machine learning algorithms is to produce a model that generalizes, that is, that predicts previously unseen observations. Overfitting occurs when a model fits the data in the training set well, while incurring larger generalization error. The reason is, models can be too complex and therefore able to memorize the training dataset, when shown an example not previously seen in the training set. Early stopping is a technique that can be used to avoid overfitting. It allows you to stop the training process after a given number of iterations if the model performance on the test set decreases or stays constant. Regularization is another commonly used technique to avoid overfitting. It reduces the model complexity by adding a penalty term to the loss function. By reducing the absolute value of the weights only smooth functions are allowed. Another form of regularization which has appeared more recently in the context of neural networks is *Dropout* which reduces the model complexity by randomly dropping neurons from the neural network during training in each iteration.

### 2.3. Conditional Generative Adversarial Networks

Generative Adversarial Networks (GANs) are classified within the group of generative models. They consist of two adversarial models: a generative model  $G$  that captures the data distribution, and a discriminative model  $D$  that estimates the probability that a sample came from the training data rather than  $G$ . The generator's  $G$  loss during training depends on the probability of the discriminative model of making an error by assigning the class *real* to *fake* data, that is, data generated by the generator. Other generative models such as Gaussian Mixture Models are mostly based on maximum likelihood estimate however maximum likelihood estimation may not represent the complexity of the actual data distribution and cannot learn the high-dimensional data distributions. The generator in a GAN model takes a fixed-length random vector  $z \sim p(z)$  as input and generates a sample  $x \sim P_{data}^*(x)$  in the domain where  $P_{data}^*(x)$  is the true distribution.

Conditional Generative Adversarial Networks are an extension of GANs where a conditional setting is applied. The generator receives a new input along the random variable  $z$  which adds extra information about the sample to be generated. Similarly the discriminator is trained to classify samples that incorporate this new information.



**Figure 4:** Neural network experiment 1.  $\chi^2$  value distribution of samples in the training set.

## 3. Data preparation and implementation

In this work we have included 37 observables. Observable is referred to a measurement of either an angular observable, a branching ratio, or a ratio, in a particular energy bin. We have included the measurements in different energy bins of the angular observables,  $P_1, P_2, P_4, P_5$ , the branching ratios  $BrK^0, BrK^{0*}$ , and the ratios  $R_K, R_{K^*}$ .

## 4. Results

### 4.1. Neural Network

#### 4.1.1. Experiment 1

We randomly generated 2500 model coefficients pairs  $(C_9, C_{10})$  by generating 50 values for  $C_9 \in [-5, 5]$  and 50 values for  $C_{10} \in [-5, 5]$  and forming the Cartesian product  $(C_9, C_{10}) : C_9 \times C_{10}$ . For each pair, all 37 predicted observable values were arranged in a 37-dimensional vector.

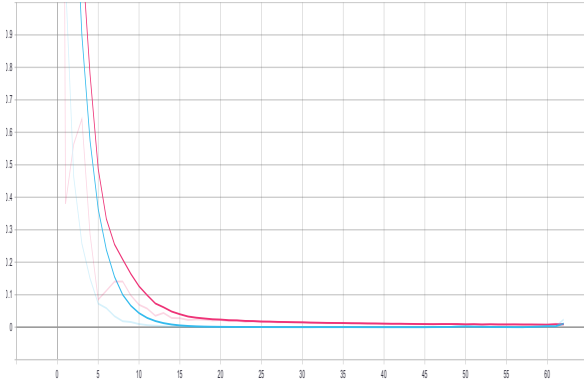
$$\begin{aligned} \mathbf{x} : & \quad \boxed{\text{Obs1}} \quad \dots \quad \boxed{\text{Obs37}} \\ \mathbf{y} : & \quad \boxed{C_9} \quad \boxed{C_{10}} \end{aligned}$$

The pair  $C_9 = -0.92$  and  $C_{10} = 0.1$  generates a sample with  $\chi^2 = 30.67$ , the lowest in the training set.

The neural network consisted of a 37-neuron input layer, one hidden layer with 2 neurons with RELU activation function and a 2-neuron output layer. We divided the original sample into a training (0.9) and a validation set (0.1). We selected a batch size of 512 and trained the model during 100 epochs with early stopping on validation mean square error equal to 10 to avoid overfitting. In figure 5 we see that the network has a low mean square error on both training and validation sets. After training the network predicts a pair  $C_9 = -0.75, C_{10} = 0.13$  with a  $\chi^2 = 30.85$ .

#### 4.1.2. Experiment 2

We randomly generated 100 model coefficients pairs  $(C_9, C_{10})$  by generating 10 values for  $C_9 \in [-5, 5]$  and 10 values for  $C_{10} \in [-5, 5]$  and computed the predicted values for each of the different energy bins. This time, to account for the



**Figure 5:** Neural Network experiment 1. Epoch mean square error training (blue) and validation (purple).

hadronic uncertainties emerging from form factor contributions, a function  $F(q^2) = a_0 + a_1 q^2 + a_2 (q^2)^2$  is added to each observable bin. The coefficients were generated by sampling from a random uniform distribution in the range  $[-0.01, 0.01]$ . Coefficients  $a_1$  and  $a_2$  were further scaled dividing by 100 and 1000 respectively. We selected 100 different coefficients  $a_0, a_1, a_2$  for each of the observables  $P_1, P_2, P_4, P_5, BrK^0, BrK^{0*}, R_K, R_{K^*}$ .

$x :$ 

Obs1	...	Obs37
------	-----	-------

$y :$ 

$C_9$	$C_{10}$	$a_{i1}$	...	$a_{i8}$
-------	----------	----------	-----	----------

where  $i = 0, 1, 2$  and  $a_{ij}$  is the  $i$  coefficient of observable  $j$ .

The neural network consisted of just one hidden layer with only two neurons. After testing different network configurations we realized that adding many neurons or layers resulted in quickly overfitting and the neural network predicting values with high  $\chi^2$  values. In figure 7 we see that the mean square error decreases for both the training and the validation sets. We use early stopping to avoid overfitting so the training process is stopped if the mean square error does not decrease after 10 iterations.

When generating the training dataset to feed the neural network we found the sample with the lowest  $\chi^2$  value with the following coefficients:

$C_9 = -0.89, C_{10} = 0.11, \chi^2 = 28.19$	$a_0$	$a_1$	$a_2$
$P_1$	0	0	0
$P_2$	-0.01	0	0
$P_4$	0	0	0
$P_5$	-0.01	0	0
$BrK^{0*}$	-0.01	0	0
$BrK^0$	0	0	0
$R_K$	0.01	0	0
$R_{K^*}$	0	0	0

After training the neural network, it predicted the following coefficients for the experimental samples:

$C_9 = -0.79, C_{10} = -0.07, \chi^2 = 29.5$	$a_0$	$a_1$	$a_2$
$P_1$	0	0	0
$P_2$	0	0	0
$P_4$	0.03	0	0
$P_5$	0.02	0	0
$BrK^{0*}$	-0.01	0	0
$BrK^0$	-0.01	0	0
$R_K$	-0.01	0	0
$R_{K^*}$	-0.01	0	0

## 4.2. GAN

### 4.2.1. GAN architecture

The GAN have been built using TensorFlow, and open-source software library developed by Google and Keras, an open-source neural-network library written in Python that allows you to build different Neural Network architectures in terms of layers. Training GAN models is hard, model parameters might oscillate, destabilize and never converge, the generator might collapse producing limited varieties of samples, the discriminator can get so successful that the generator gradient vanishes and learns nothing, an unbalance between the generator and discriminator produces overfitting and last but not least GAN are highly sensitive to hyper-parameter selections. In this work we have selected the following architecture:

### 4.2.2. Experiment 1

In this experiment we trained the GAN on the basic training dataset. Figure 4 shows the histogram of the training dataset  $\chi^2$  values. The minimum value is  $\chi^2 = 7.6$ . Once the GAN was trained we generated 100000 random samples from it and selected the sample with the minimum  $\chi^2$  value. In figure 8 we plotted this sample along the experimental bin values.

### 4.2.3. Experiment 2.

In this experiment we added to each of the samples in the basic training dataset the  $C_9, C_{10}$  coefficients that generated the sample. The  $\chi^2$  histogram should be the same as in experiment one, we just concatenated the  $C_9, C_{10}$  to the training samples. We trained the GAN on this new dataset and after training generated 100000 samples and selected the one with the lowest  $\chi^2$  value. In this experiment the minimum value obtained was 10.2.

Obs1	...	Obs37	$C_9$	$C_{10}$
------	-----	-------	-------	----------

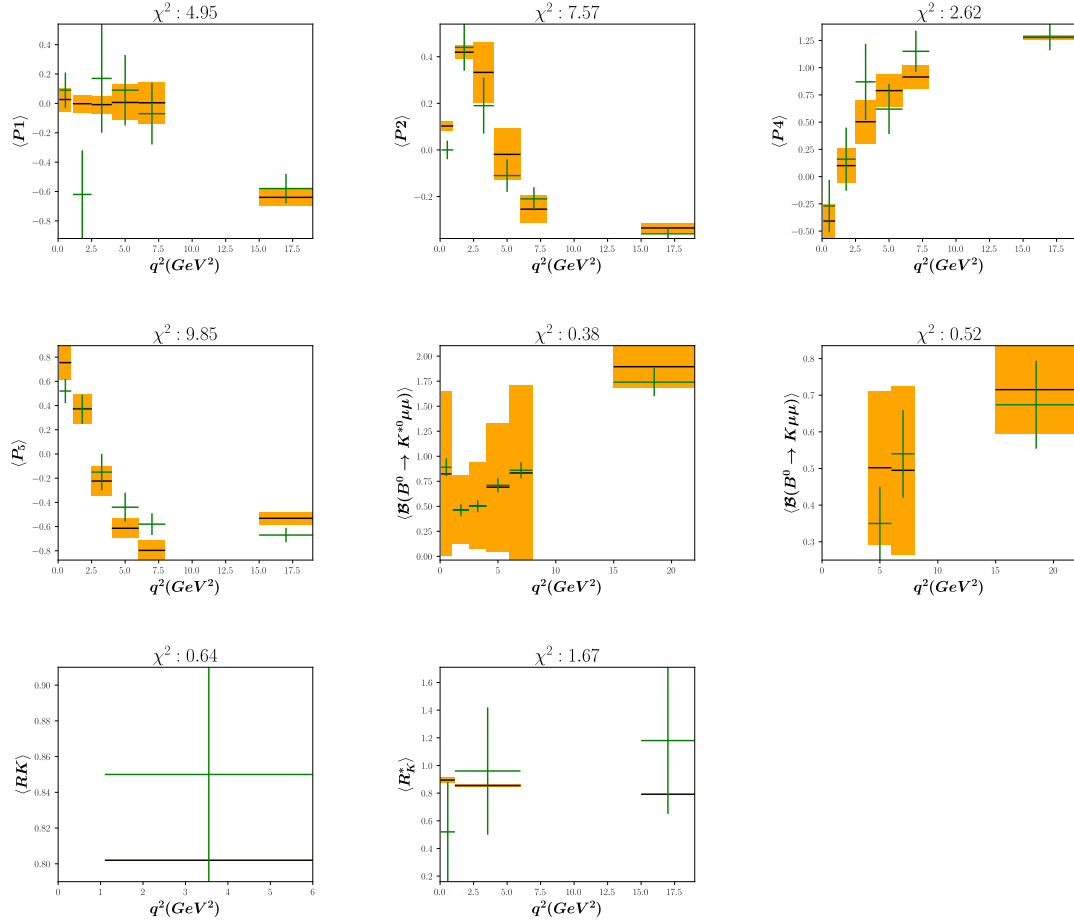
## 5. Conclusions

### CRediT authorship contribution statement

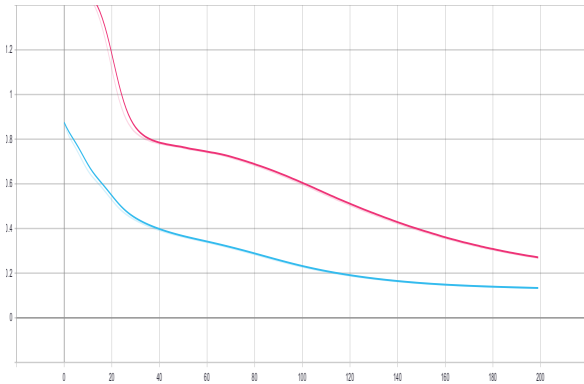
**Óliver Partida:** Conceptualization of this study, Methodology, Software.

## References

- [1] Algueró, M., Capdevila, B., Crivellin, A., Descotes-Genon, S., Masjuan, P., Matias, J., Vito, J., 2019. Emerging patterns of new physics with and without lepton flavour universal contributions. The

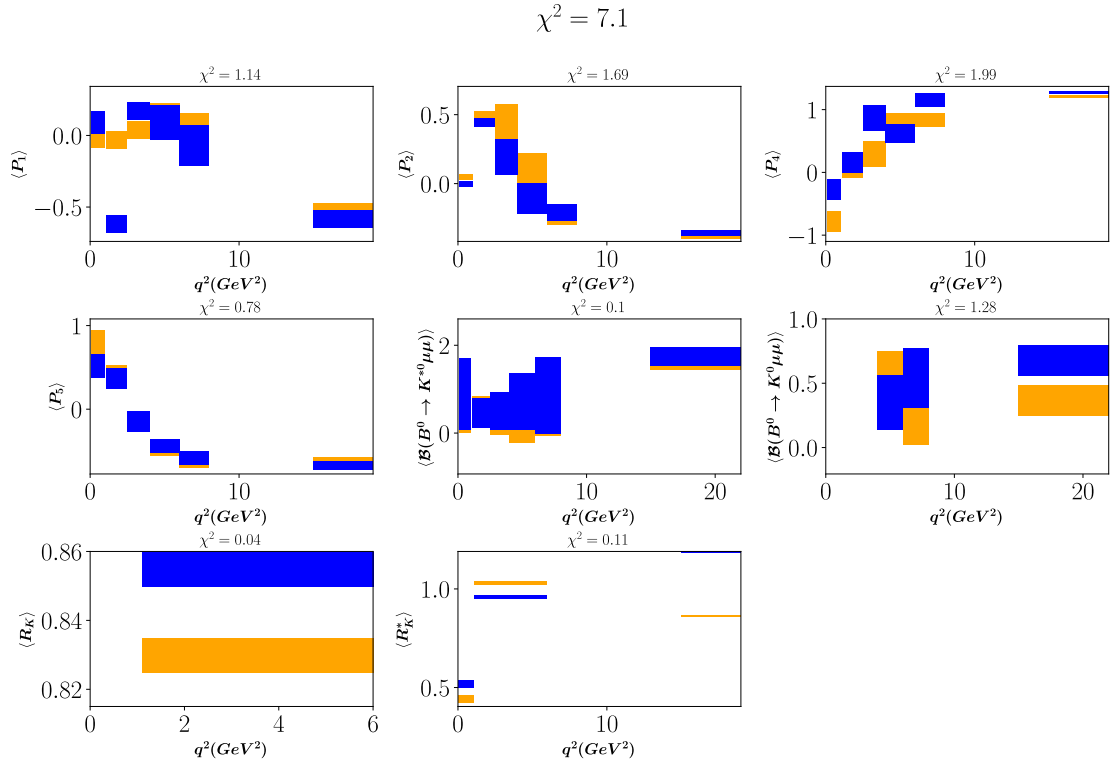
Model: C9: -0.89, C10: 0.11,  $\chi^2$  : 28.19


**Figure 6:** Sample with lowest  $\chi^2 = 28.19$  (orange) and experimental bin values (blue). This sample was generated by randomly selecting coefficients. The network found a very similar result with  $\chi^2 = 29.5$



**Figure 7:** Neural Network experiment 2 (All). Mean square error training (blue) and validation (purple).

- [2] Capdevila, B., Crivellin, A., Descotes-Genon, S., Matias, J., Virto, J., 2017. Patterns of new physics in  $b \rightarrow s \ell^+ \ell^-$  transitions in the light of recent data. arXiv:1704.05340.
- [3] Descotes-Genon, S., Hofer, L., Matias, J., Virto, J., 2016. Global analysis of  $b \rightarrow s$  anomalies. Journal of High Energy Physics 2016. URL: [http://dx.doi.org/10.1007/JHEP06\(2016\)092](http://dx.doi.org/10.1007/JHEP06(2016)092), doi:10.1007/jhep06(2016)092.



**Figure 8:** Generated sample with minimum  $\chi^2 = 7.1$  (orange) and experimental bin values (blue).