Machine Learning for New Physics Searches in $B \to K^{*0}\mu^+\mu^-$ Decays

S. Dubey[1,*] *T. E.* Browder[1], *S.* Kohani[1], *R.* Mandal[2], *A.* Sibidanov[1], *R.* Sinha[1], and *S.E.* Vahsen[1]

[1]University of Hawai'i at Mānoa , Honolulu, HI 96822, USA
[2]Indian Institute of Technology Gandhinagar, Gujarat 382055, India

Abstract. We report the status of a neural network regression model trained to extract new physics (NP) parameters in Monte Carlo (MC) simulation data. We utilize a new EvtGen NP MC generator to generate $B \to K^{*0}\mu^+\mu^-$ events according to the deviation of the Wilson Coefficient C_9 from its SM value, δC_9. We train a three-dimensional ResNet regression model, using images built from the angular observables and the invariant mass of the di-muon system, to extract values of δC_9 directly from the MC data samples. This work is intended for future analyses at the Belle II experiment but may also find applicability at other experiments.

1 Introduction

The process $B \to K^*\ell^+\ell^-$ with $\ell = e, \mu$ proceeds via a beauty-to-strange quark ($b \to s$) flavor-changing neutral current, which is forbidden at tree-level in the Standard Model (SM) of particle physics but allowed at second order [1], and is therefore sensitive to beyond the Standard Model (BSM) physics.

There are hints of new physics beyond the SM in the observed angular distributions of $B \to K^*\ell^+\ell^-$. These can be more clearly identified in angular asymmetries, such as the forward-backward asymmetry ($A_{\rm FB}$), S_5, and others, described in Ref. [2]. It is possible that these angular asymmetries are lepton-flavor violating (LFV), with this possibility first directly explored in a 2021 Belle analysis [3]. It is also possible that BSM physics is lepton-flavor universal (LFU). In the future, determining the scenario from which these apparent anomalies originate — via SM interactions with unaccounted-for hadronic effects, or BSM physics — is a key experimental problem.

We develop a new Monte Carlo (MC) model [4] for the EvtGen package and use that to produce "images" that are employed to train our neural network model. Our model is a three-dimensional, 34-layer, Residual Neural Network (ResNet) [5, 6] model trained to perform regression to extract Wilson Coefficient (C_i) [4, 7] information, $\delta C_i \equiv C_i^{\rm BSM} - C_i^{\rm SM}$, directly from data. Hence, we recast the C_9 fitting problem as a computer vision problem. To this end, we employ an MC simulation model to generate $B^0 \to K^{*0}\mu^+\mu^-$ events, according to various BSM scenarios parameterized by Wilson Coefficients. From the resulting distributions we create "quasi-images" , which are then used to train a ResNet to perform a regression task.

*e-mail: sdubey@hawaii.edu

Our approach differs from the usual application of artificial intelligence methods in HEP, which involves classification to distinguish various categories of events such as signal versus background; jet classification; or particle identification (determining particle species) etc [8]. In contrast, we will perform regression and extract a continuous parameter from data. This is similar to “fitting", a different but essential part of HEP analysis. Our neural network model learns the correlation between these images and their δC_9 labels, which is equivalent to learning the mapping between the distributions and the δC_9 labels. We apply regression and extract the parameter δC_i, where $\delta C_i = 0.0$ is the SM case. Our MC result is an example of extracting physics parameters directly from detector data. We note that the method presented here may be find broad applicability, even in the absence of LFV BSM physics.

2 Monte Carlo Simulation Model

Recently, we have implemented effective field theory couplings in a new MC generator [4] in the EvtGen framework [9]. The new MC generator uses the operator product expansion formalism in terms of Wilson Coefficients (WCs, which encode high energy/short distance physics information) C_7, C_9, C_{10}, C'_7, C'_9, and C'_{10}, where the latter three primed WCs correspond to right-handed couplings (the weak couplings in the SM are usually left-handed).

Our EvtGen model is parameterizable in terms of the WCs' deviation from their SM values, i.e. δC_i. Each of the δC_i can be chosen by the user [4]. Choosing a non-zero δC_i has the effect of altering the correlations between four variables: $q^2 \equiv M(\ell^+\ell^-)$, the cosine of the lepton helicity angle $\cos(\theta_\ell)$, the cosine of the helicity of the K^* $\cos(\theta_{K^*})$, and the angle χ between the decay planes of the di-lepton and K^* decay planes. Figure 1 shows the decay topology and the full set of angular observables.

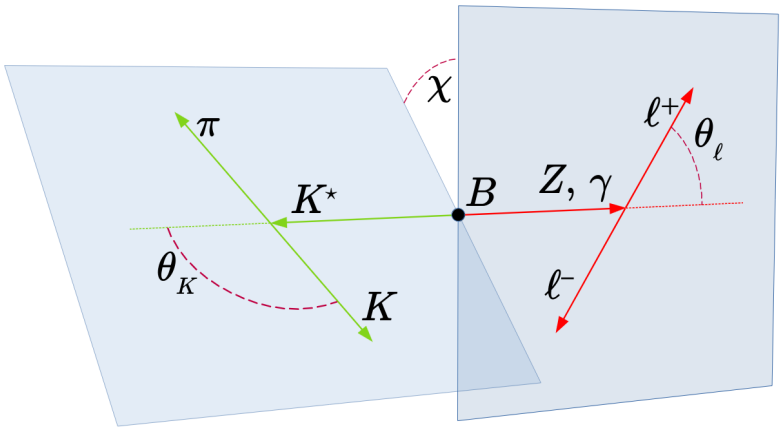


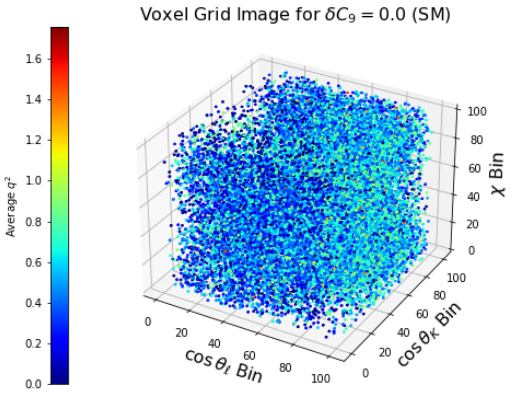
Figure 1: The $B \to K^*\ell^+\ell^-$ decay topology showing the observables [4]. For this study we only consider the di-muon channel where $\ell = \mu$.

3 Creating the Images

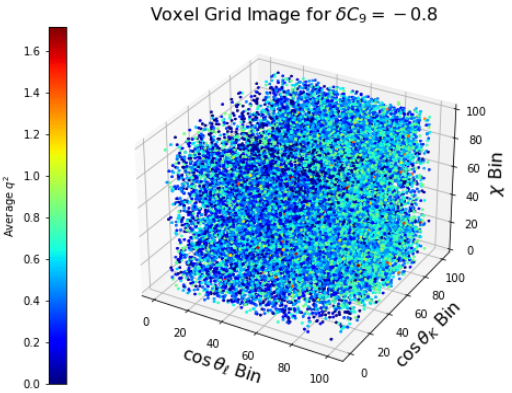
Samples of MC events are generated with $\delta C_9 \in [-2.0, 0.0]$, as global theory fits for the dimuon-specific δC_9 appear to favor a negative value near -0.9 [10]. Twenty-two values are chosen in the above range and 1×10^6 events are generated for each of the δC_9 values. Approximately 2.4×10^4 $B^0 \to K^{*0}\mu^+\mu^-$ generator-level-only events populate a single image.

This corresponds to approximately 250 ab^{-1}-equivalent integrated luminosity at a Belle II upgrade (assuming the Belle signal reconstruction efficiency, five times the Belle II target integrated luminosity.

Images are produced by binning the q^2 value of each event in bins of $\cos(\theta_\mu)$, $\cos(\theta_{K^*})$, and χ. There are 100 bins in each angular variable. Therefore, the shape of the input image is (height, width, depth) = (100, 100, 100). In effect, we have created a grid of voxels (3D pixels). We treat the q^2 values as grayscale values, so that the images are input as tensors to the neural network and each have shape (100, 100, 100, 1), where the value 1 denotes the number of color channels. Figure 2 shows two examples of these images for different values of δC_9.



(a) Voxel grid image for $\delta C_9 = 0.0$ (SM)



(b) Voxel grid image for $\delta C_9 = -0.8$

Figure 2: Voxel grid images used for training and evaluation of the ResNet. Each angular bin is a range of angular values. The angular range is divided into 100 equal-width bins. Examples for the cases of $\delta C_9 = 0.0$ (SM) and $\delta C_9 = 0.8$ are shown. The color of the voxels does not indicate that the image has multiple color channels and is only used for visualization.

4 The Neural Network

As discussed above, a ResNet variation of the CNN is employed. Specifically, a three-dimensional variation of the ResNet studied in Ref. [5] is used and is built using TensorFlow [11] and Keras [12]. There are 34 convolutional layers in the main neural network path, utilizing the ReLU activation function [6]. Stochastic gradient descent is used for optimization. The loss function is the mean absolute error (MAE) [6]. At the end of network there is one fully-connected layer employing 5000 neurons, followed by a drop-out layer with a 50% drop-out probability. The final layer is a dense layer with one neuron and a linear activation function that performs the regression task to extract δC_9 values directly from the images [1]. No hyperparameter optimization was done as the initial model appeared to performed well.

[1] The linear activation function is used as its range is a continuum of values in $(-\infty, \infty)$.

4.1 Training the Neural Network

Approximately 12000 voxel grid images are used for training, corresponding to about 540 images for each δC_9 value. This training set is split with approximately 20% reserved for validation. To facilitate learning, the learning rate is reduced every five epochs by a factor of 1/5, if no improvement in the validation loss is seen. Early stopping is implemented if there is no improvement in the validation loss (MAE) after 50 epochs. Training is performed using the GPU nodes of the University of Hawai'i's MANA HPC cluster.

5 Results

As this is a AI/ML model for regression and not classification, standard tools to assess the trained model used in classification, e.g. the receiver operating characteristic (ROC) curve, are not applicable here. Instead to test the model, we examine ensembles of MC simulation experiments. For each of the 22 WCs that were used to generate training images, statistically independent samples of 1600 images are generated for testing. Each of the ensembles of 1600 images are passed through the trained network and a distribution of predicted δC_9 values are obtained. These distributions are fitted with a Gaussian function and the mean and width from these fits are used to assess performance. We also perform this test using δC_9 values that are between the ones used to generate the training images. This provides a further test of model robustness.

These ensembles tests are performed for all δC_9 values used to generate images for training and for δC_9 values between the ones used to generate images that were not used in training. The fit results are plotted against their generated values to obtain a linearity plot, shown in Fig. 3.

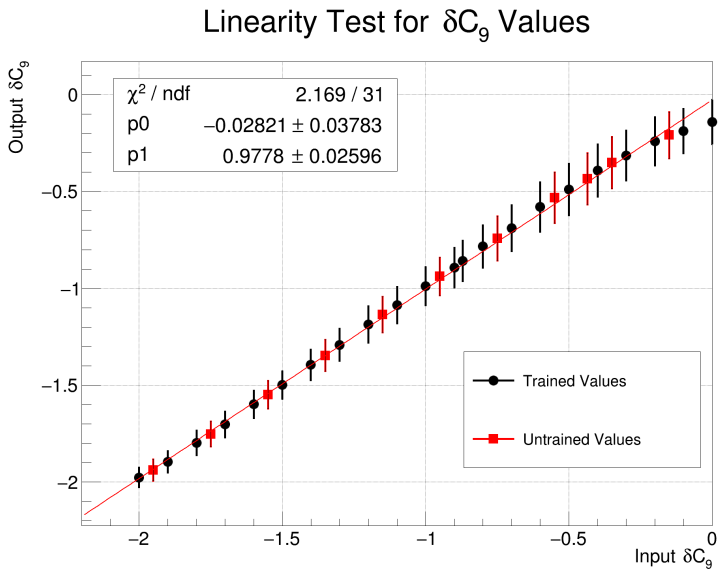


Figure 3: From ensemble experiments, it is seen that the trained ResNet is able to correctly extract the different δC_9 values from independent and unlabeled images. The black points are from experiments where the images are generated according to δC_9 values the ResNet has been trained with, and the red points are from experiments where the images are generated according to δC_9 values with which the ResNet has *not* been trained.

6 Discussion

As seen in the linearity plot in Fig. 3, the ResNet appears to be able to obtain a mapping between δC_9 values and MC signal events when those events are recast into images. However, as one approaches the SM $\delta C_9 = 0.0$ value, performance degrades since the images closer to the SM are harder to distinguish. Further, when compared to the results of the 4D unbinned maximum likelihood fit to generator-level signal MC samples in Ref. [4], the error bars from the ensemble test for $\delta C_9 = 0.0$ can be 60% larger, at 250 ab^{-1}-equivalent signal events. Nevertheless, we do not interpret this as a major defect for the method presented here. These issues are likely to be ameliorated with more training data and an extension of the training data to the positive δC_9 range, so that the model can better learn the mapping. We expect further improvements by separating B and $\bar{B}$ events into separtate images, and removing multiple entries in bins, which constitute approximately 3% of all bins.

Further, using a 4D unbinned maximum likelihood fit has a number of problems in a real high-energy physics experiment. In the presence of backgrounds and resolutions, it is difficult to parameterize the backgrounds, efficiency, and resolution in multiple dimensions. These associated issues can be greatly mitigated if the task is recast as a computer vision problem.

The main issue with the method described here is one of computational power and storage. We used δC_9 and high-statistics generator-level MC samples for the demonstration. In reality, a fully trained and useful model would have to be trained using images according to all the WCs mentioned above, as well as different integrated luminosities (assuming applicability at Belle II).

7 Conclusion

We have trained a three-dimensional ResNet to learn a mapping between different δC_9 values and images created using MC simulations of $B \to K^* \mu^+ \mu^-$ decays. We have recast the problem of fitting complicated multi-dimensional distributions using maximum likelihood techniques as a standard computer vision problem. This should make it easier to take into account experimental complexities such as backgrounds and experimental resolutions, and does not require projecting down to a lower dimension (e.g. in this case to angular asymmetries such as A_{FB} and S_5), losing potentially valuable information. Our approach may also find application to studies of $\bar{B}^0 \to D^{*+} \ell^- \bar{\nu}$, where a new BSM physics generator has also recently been developed [13].

It has been shown that a ResNet is indeed able to learn this correlation and successfully extract information about the relevant physics parameters. Difficulties with this method will likely be mitigated with increased training sample sizes and additional computational resources. We will improve and publish results after the generation of an enhanced training set.

8 Acknowledgements

We thank our colleagues on Belle II as well as the KEK computing group for their excellent operation of the KEK computing center. We have used the University of Hawai‘i MANA HPC cluster. The technical support and advanced computing resources from University of Hawai‘i Information Technology Services – Cyberinfrastructure, funded in part by the National Science Foundation CC* awards # 2201428 and # 2232862 are gratefully acknowledged. We also thank Hongyang Gao (ISU) for his seminal suggestion to use computer vision techniques to search for new physics couplings, and Chunhui Chen (ISU) for facilitating the meeting, as

well as Peter Sadowski (UHM) and Jeffrey Schueler (UNM) for their helpful discussions on machine learning and technical advice.

References

[1] S.L. Glashow, J. Iliopoulos, L. Maiani, Phys. Rev. D **2** (1970)
[2] R. Aaij et al., Journal of High Energy Physics **2017**, 55 (2017)
[3] S. Wehle et al. (Belle Collaboration), Phys. Rev. Lett. **126**, 161801 (2021)
[4] A. Sibidanov et al., *Detecting lepton universality violation in angular distributions of* $B \to K^* \ell^+ \ell^-$ *decays* (2022), 2203.06827
[5] K. He, X. Zhang, S. Ren, J. Sun, *Deep residual learning for image recognition* (2015), 1512.03385
[6] A. Geron, *Hands-on machine learning with scikit-learn, keras, and TensorFlow*, 2nd edn. (O'Reilly Media, Sebastopol, CA, 2019)
[7] M.D. Schwartz, *Quantum Field Theory and the standard model* (Cambridge University Press, 2013)
[8] HEP ML Community, *A Living Review of Machine Learning for Particle Physics, and references therein*, https://iml-wg.github.io/HEPML-LivingReview/
[9] D.J. Lange, Nucl. Instrum. Meth. A **462**, 152 (2001)
[10] W. Altmannshofer, P. Stangl, The European Physical Journal C **81**, 952 (2021)
[11] M. Abadi et al., *TensorFlow: Large-scale machine learning on heterogeneous systems* (2015), software available from tensorflow.org, https://www.tensorflow.org/
[12] F. Chollet et al., *Keras*, https://keras.io (2015)
[13] B. Bhattacharya et al., Phys. Rev. D **107**, 015011 (2023)