First block exercises

Author: Aleix López Pascual
Statistics and Data Analysis

December 2, 2017

Abstract: The angular distribution of the electrons produced in muon decays ($\mu^- \to e^- + \nu_\mu + \bar{\nu}_e$) is described by some known distribution. This theory predicts the probability for an event to occur as a function of the scattering angle θ (random variable) and the muon polarisation P_μ (physical parameter). The main purpose of our exercises is to estimate P_μ. We are going to do it in two different ways: computing the basic statistical momentums (exercise 2) and finding the most probable value of P_μ using conditional probabilities (exercise 7). Furthermore, we are going to elaborate some kind of checks, in order to verify what we are doing (Law of large numbers, Student's t-distribution, CLT and Pearson's χ^2 test).

Exercise 1

A probability density function (pdf) gives us the probability of observing a value within an infinitesimal interval $[x, x+dx]$, where x is a single continuous random variable. A pdf must satisfy three properties in accordance to the Kolmogorov axioms of probability:

1. $f(x) \geq 0 \ \ \forall x \in \Omega$
2. $\int_{x\in\Omega} f(x)dx = 1$
3. $\mathrm{P}(x \in [x_i, x_i+dx_i] \cup [x_j, x_j+dx_j]) = \mathrm{P}(x \in [x_i, x_i+dx_i]) + \mathrm{P}(x \in [x_j, x_j+dx_j])$ where x_i, x_j are mutually exclusive events

The function we want to prove that it is a pdf is given by

$$\frac{d\Gamma}{d\cos\theta} = \frac{1}{2}(1 - \frac{1}{3}P_\mu \cos\theta) \qquad P_\mu \in [-1, 1] \tag{1}$$

where P_μ is the muon polarisation and $\cos\theta$ is the angle between the electron and the muon polarisation vector in the muon rest frame. First of all, we should note that we can use $\theta \in [0, 2\pi]$ or $\cos\theta \in [-1, 1]$ as a random variable. Both give the same result. However, we prefer to use $\cos\theta$ in order to make the pdf simpler. We proceed with the demonstrations:

1. It is clear that the minimum value of the function is reached when $P_\mu \cos\theta = 1$. Then $f(P_\mu \cos\theta = 1) = \frac{1}{2}(1 - \frac{1}{3}) = \frac{1}{3} > 0$. This can also be seen plotting the function (Fig. 1).
2. We denote $x = \cos\theta$
   $\int_\Omega f(x)dx = \int_{-1}^1 \frac{1}{2}(1 - \frac{1}{3}P_\mu x)dx = \frac{1}{2}\big[\int_{-1}^1 dx - \frac{1}{3}P_\mu \int_{-1}^1 xdx\big] = 1$
3. $\mathrm{P}(x \in [x_i, x_i+dx_i]) + \mathrm{P}(x \in [x_j, x_j+dx_j]) = \int_{x_i}^{x_i+dx_i} f(x)dx + \int_{x_j}^{x_j+dx_j} f(x)dx = \int_{x_i}^{x_i+dx_i} \frac{1}{2}(1 - \frac{1}{3}P_\mu x)dx + \int_{x_j}^{x_j+dx_j} \frac{1}{2}(1 - \frac{1}{3}P_\mu x)dx = \frac{1}{2}[dx_i - \frac{1}{6}P_\mu((x_i+dx_i)^2 - (x_i)^2)] + \frac{1}{2}[dx_j - \frac{1}{6}P_\mu((x_j+dx_j)^2 - (x_j)^2)] = \frac{1}{2}[dx_i - \frac{1}{3}P_\mu x_i - \frac{1}{3}P_\mu dx_i^2 + dx_j - \frac{1}{3}P_\mu x_j - \frac{1}{3}P_\mu dx_j^2] = \frac{1}{2}[(dx_i + dx_j) - \frac{1}{3}P_\mu(x_i dx_i + x_j dx_j)] = \mathrm{P}(x \in [x_i, x_i+dx_i] \cup [x_j, x_j+dx_j])$

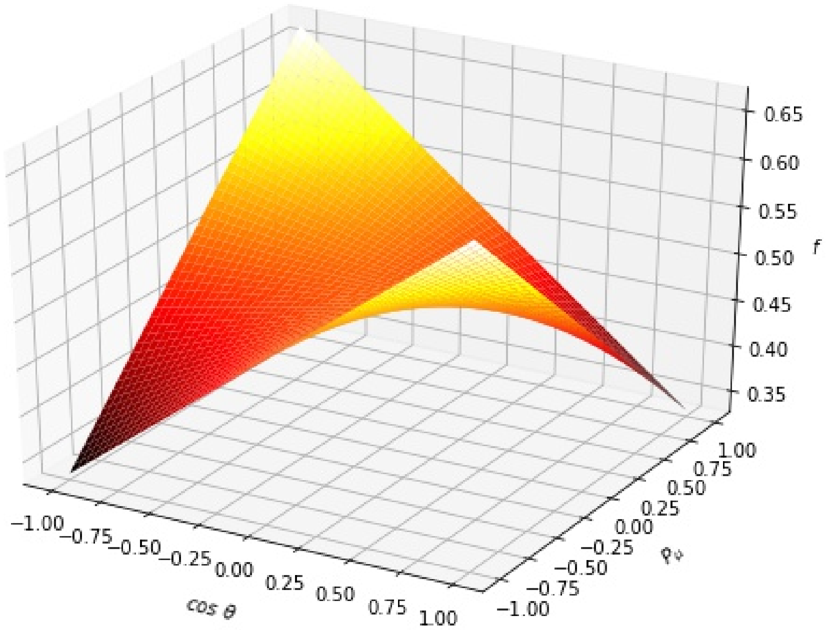


Figure 1: Normalised differential cross-section f as a function of $\cos\theta$ and P_μ

Exercise 2

We are going to use the Monte Carlo method to simulate our experimental data. This is a common process in particle physics, which is called event generator. The motivation of this is that we want to compute the basic first momentums of the distribution. As we can't identify this pdf as any familiar pdf, we cannot just do it directly (in concrete, our pdf is a special case of the β-function). Therefore, we have two options: calculate the momentums analytically, which in general is complicated, or simulate the data according to our pdf and then estimate the momentums numerically. Note that in our case the analytical way is trivial, but when we have multidimensional problems, the Monte Carlo method becomes truly useful.

The Monte Carlo algorithm consist in generating random values generated according to a uniform distribution $\mathcal{U}(0,1)$ and then determine a sequence of values distributed according to the pdf $\frac{d\Gamma}{d\cos\theta}$. In order to generate the pseudo-random numbers, we could make our multiplicative linear congruential algorithm. However, it is not necessary since we already have a module, so-called numpy.random, which implements pseudo-random number generators for various distributions. We have performed two different Monte Carlo methods:

The inverse transform method

Here the task is to find a function $x(r)$ that is distributed according to the specified pdf $f(x)$, given that r follows a uniform distribution $\mathcal{U}(0,1)$. This function must satisfy

$$F(x(r)) = G(r) \Rightarrow$$
$$\int_{-\infty}^{x(r)} f(x')dx' = \int_{-\infty}^{r} g(r')dr' = r$$

We substitute $f(x')$, then we integrate and solve for $x(r)$. We obtain

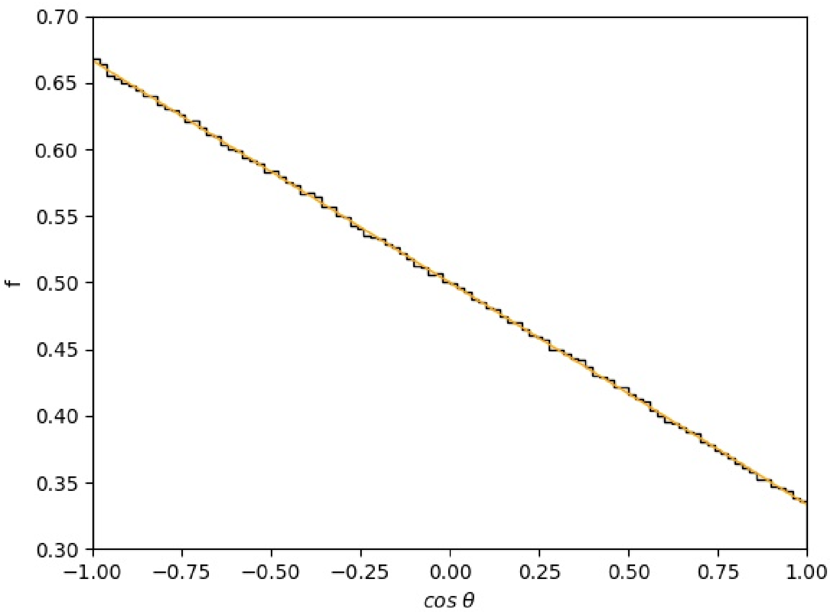
$$x(r) = \frac{3}{P_\mu}\left[1 \mp \sqrt{1 + \frac{2}{3}P_\mu\left(\frac{P_\mu}{6} + 1 - 2r\right)}\,\right] \tag{2}$$

$x(r)$ is a function of a random variable, so it's a random variable too, distributed according to $f(x)$.

In order to reconstruct the pdf we plot the obtained values $x(r)$ in a histogram, which takes into account the number of occurrences of x in subintervals (bins). As you can observe, we must impose a value for P_μ to proceed. Moreover, note that we have found two solutions for $x(r)$. Once we plot the histogram for each one, we realise that only that generated by the minus sign in (Eq. 2) gives us the correct fitting. We compare this result with the original pdf (Fig. 2).

The acceptance-rejection method

In case the form of $F(x)$ is too complex to work with, we have this case. We generate a random number $x \in \mathcal{U}(-1, 1)$ in accordance to our random variable $\cos\theta$. Then we generate a second random number $u \in \mathcal{U}(0, f_{max})$. Finally, if $u < f(x)$, we accept x. If not, we reject x and repeat.



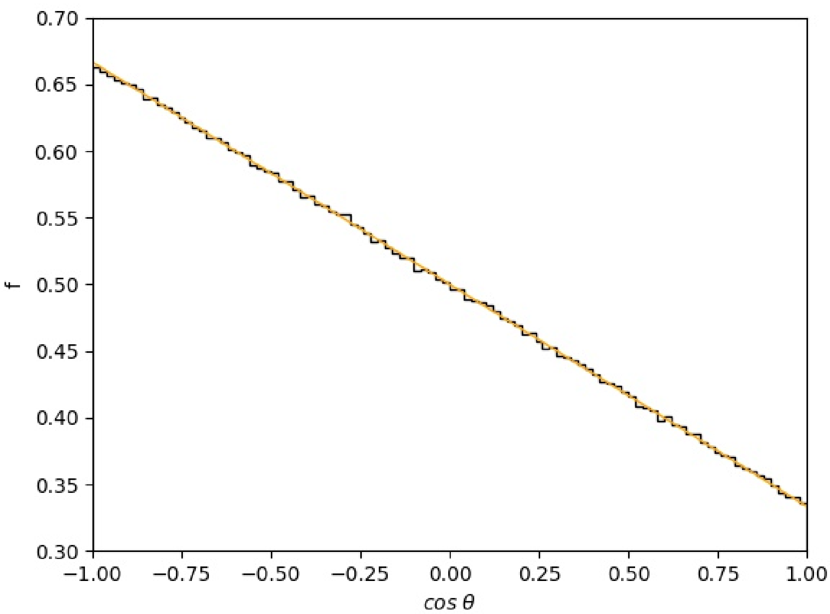


Figure 2: Histogram using the inverse transform method (top) and using the acceptance-rejection method (bottom) compared with the pdf. Number of measurements = 10000000. Number of bins = 100. $P_\mu = 1$.

The accuracy of both methods increases with the number of measurements. Note that the number of bins cannot be very large in comparison to the number of measurements. Otherwise, the histogram does not fit the pdf. As we can observe both mechanisms works fine. However, it is important to mention that while for the inverse transform method we get the same number of measurements than the initial number of events, in the acceptance-rejection method we obtain less since we do not accept all points.

Mean, variance, skewness and kurtosis

Now we are going to calculate the basic statistical momentums from the data generated with the Monte Carlo. It is clear that we are estimating parameters. We can define our own functions to do it in accordance to the definitions of the estimators. However, we already have modules (numpy, scipy.stats) which do it directly.

Mean	Variance	Skewness	Kurtosis
-0.11104	0.3209928	0.22881	-1.111632
-0.11085	0.3209499	0.22940	-1.111305

Table 1: Basic statistical momentums of the distribution based on the Monte Carlo. Fisher's definition has been used to compute the kurtosis. Inverse transform method (top) and acceptance-rejection method (bottom).

As we can see, we obtain values for the skewness and kurtosis different from 0, since our pdf is not normal. It has a positive skew because of the mass distribution concentrated on the left of the figure. It has negative kurtosis in accordance to the flatness of the pdf. We have computed it with both methods. We obtain very similar estimations.

Exercise 3

We know from the theory of our muon decay study that the mean of $\frac{d\Gamma}{d\cos\theta}$ depends on the polarisation P_μ as $\mu = -P_\mu/9$. In fact, we can verify this dependency by doing various experiments (Monte Carlos) of N measurements for different values of P_μ and computing the mean for each one.

P_μ	$\lvert P_\mu - \hat{P}_\mu \rvert$	P_μ	$\lvert P_\mu - \hat{P}_\mu \rvert$
-1.0	0.00014	0.2	0.00030
-0.8	0.00082	0.4	0.00159
-0.6	0.00241	0.6	0.00079
-0.4	0.00019	0.8	0.00022
-0.2	0.00056	1.0	0.00288
0.0	0.00035		

Table 2: Discrepancy between $\hat{P}_\mu = -9\hat{\mu}$ and the chosen P_μ

As we can observe the dependency is satisfied. Remember that the purpose of our study is to compute the polarisation of the muons P_μ. Given this relationship, it is clear that we can estimate $\hat{P}_\mu$ just computing $\hat{\mu}$.

We have seen we can estimate the parameter P_μ, but now we want to give some measure of the statistical uncertainty of the estimation. That is, if we repeat the entire experiment for a given fixed P_μ a large number of times with N measurements each time, each experiment would give a different estimated value for the parameter P_μ. How widely spread will they be? We define this with the variance of the estimator.

In order to compute the variance using the Monte Carlo method, first we need to do an experiment of 100000 measurements using a "true" value $P_\mu = 0.5$. We obtain $\hat{P}_\mu = -9\hat{\mu} = 0.5187085$. Regarding this Monte Carlo as the "real" one, we perform 1000 further experiments with 100000 measurements each using the $\hat{P}_\mu$ of the first experiment as the true value. Once we have gathered these several values of $\hat{P}_\mu$, we compute the variance using the module numpy.

The variance obtained using 1000 experiments is $Var(\hat{P}_\mu) = 0.000305$. Then using $\sigma_{\hat{P}_\mu} = \sqrt{Var(\hat{P}_\mu)}$, we can finally report our estimator as

$$\hat{P}_\mu = 0.519 \pm 0.017$$

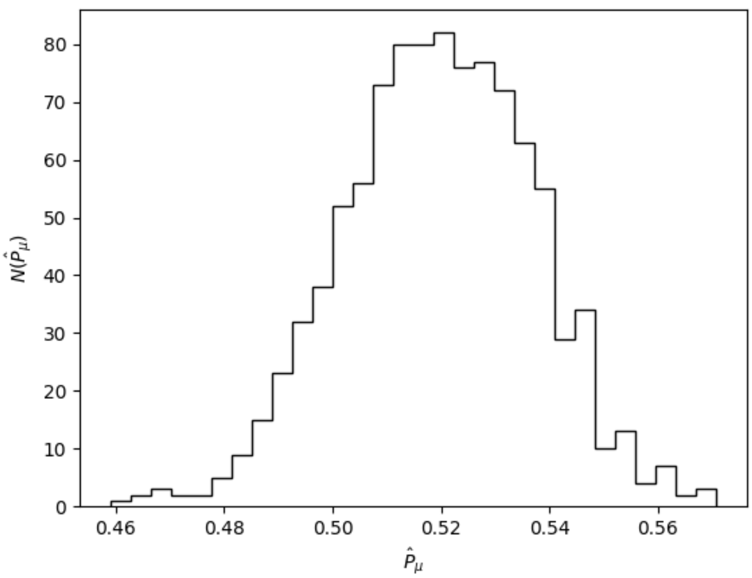


Figure 3: Histogram of the estimator $\hat{P}_\mu$ from 1000 Monte Carlo experiments with 100000 measurements per experiment. Number of bins = 30. Monte Carlo "true" parameter $P_\mu = 0.519$.

Exercise 4

We are going to verify that our experiment satisfies the law of large numbers, which concerns the convergence of the average (sample mean). According to the law, the average of the results obtained from a large number of trials should be close to the expected value (mean), and will tend to become closer as more trials are performed. Therefore, this law "guarantees" stable long-term results for the averages of some random events.

In order to be able to apply the law, we must assume a sequence of independent random variables, each having the same mean and variance. In our case, the random variables generated from the Monte Carlo method satisfy this condition, since they are generated from the same pdf and all are mutually independent.

As we have seen, $\hat{P}_\mu = -9\hat{\mu}$ and we know that the sample mean is an estimator for the mean, so it is clear that the law of large numbers will also affect $\hat{P}_\mu$. An illustration of this it is shown in Fig. 4 and Fig. 5.

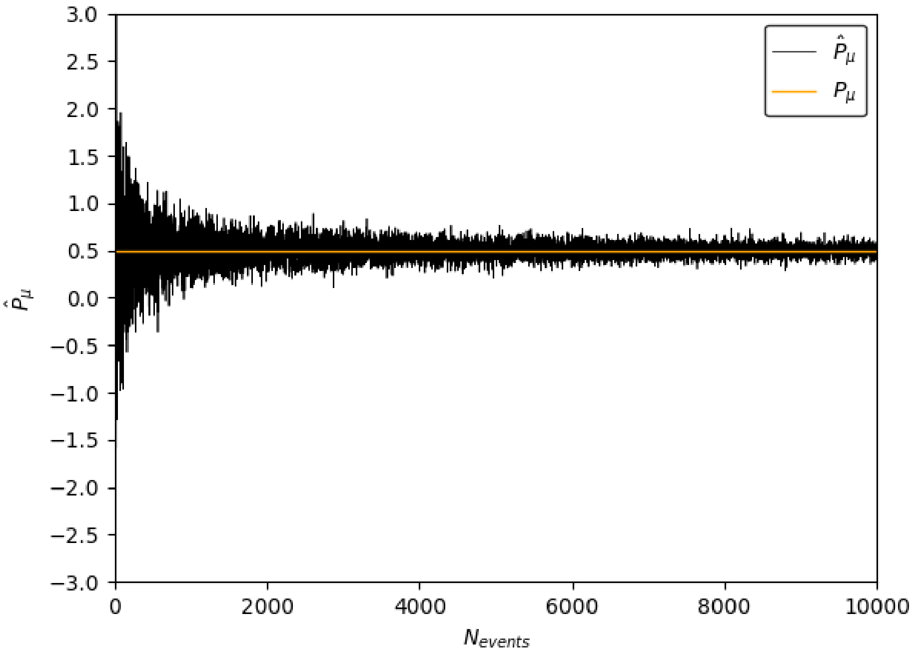


Figure 4: Illustration of the law of large numbers for $\hat{P}_\mu$ at small scale. The number of measurements has been increased to 10000 by steps of 1. A Monte Carlo has been performed for each step. $P_\mu = 0.5$.

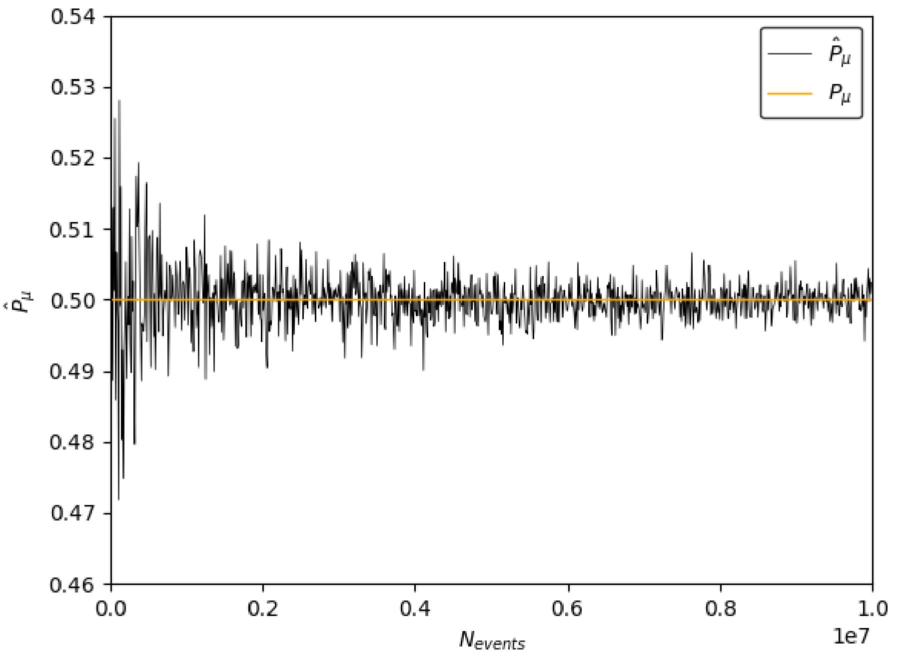


Figure 5: Illustration of the law of large numbers for $\hat{P}_\mu$ at large scale. The number of measurements has been increased to 10^7 by steps of 10000. A Monte Carlo has been performed for each step. $P_\mu = 0.5$.

Exercise 5

Student's t-distribution

We want to test if the sample mean $\bar{X}$ of our experiments is consistent with the theoretical value μ. In order to do this we are going to compute the so-called Student's t-variable defined as

$$t = \frac{\sqrt{N}(\bar{X} - \mu)}{s} \tag{3}$$

where X is the sequence of independent random variables, N is the number of events of the experiment, μ is the "true" value that we are going to take as $\mu = -P_\mu/9$ and s^2 is the unbiased estimate of the variance, which we can compute with numpy.var(ddof=1). Then we are going to verify that t follows the Student's t-distribution. As we need to have enough data to be able to identify some distribution, we must perform several experiments (Monte Carlos) with the same number of events and compute t for each one of them. Finally we generate a histogram for the t values and compare it with the theoretical pdf (Fig. 6). We have generated the theoretical pdfs using scipy.stats.t with the parameter $\nu = N - 1$ where N is the number of events and scipy.stats.norm. It is important to highlight that if we use our acceptance-rejection method here, we do not obtain a fit as good as if we use the inverse transform method, since in the first one not all the measurements are accepted.

	Mean	Variance	Skewness	Kurtosis
Theory N=100	0	1.02	0	0.06
MC N=100	0.006	1.02	-0.0005	0.08
Theory N=1000	0	1.002	0	0.006
MC N=1000	-0.0009	0.9999	-0.03	0.1

Table 3: Comparison of the basic statistical momentums of the t-distribution based on the Monte Carlo and based on the theory. Fisher's definition has been used to compute the kurtosis.

Central Limit Theorem

Here we want to validate that our experiments also satisfy the CLT, which should be the case since we are dealing with independent random variables. This theorem states that when independent random variables are added, their properly normalized sum tends toward a standard normal distribution even if the original variables themselves are not normally distributed. As we can observe from Fig. 6 and Table 3, the Student's t-distributions approaches the standard normal distribution $\mathcal{N}(0,1)$ when $N \to \infty$ as predicted by the CLT.

Furthermore, we can also verify the CLT in such a way similar to the previous case. We now compute Gaussian variables defined as

$$g = \frac{\sum_{i=1}^{N} X_i - \sum_{i=1}^{N} \mu_i}{\sqrt{\sum_{i=1}^{N} \sigma_i^2}} = \frac{\sqrt{N}(\bar{X} - \mu)}{S} \tag{4}$$

where S^2 is the bias sample variance. Then, we are going to verify that g follows the normal standard distribution when the number of events $N \to \infty$ as predicted by the CLT (Fig. 7).

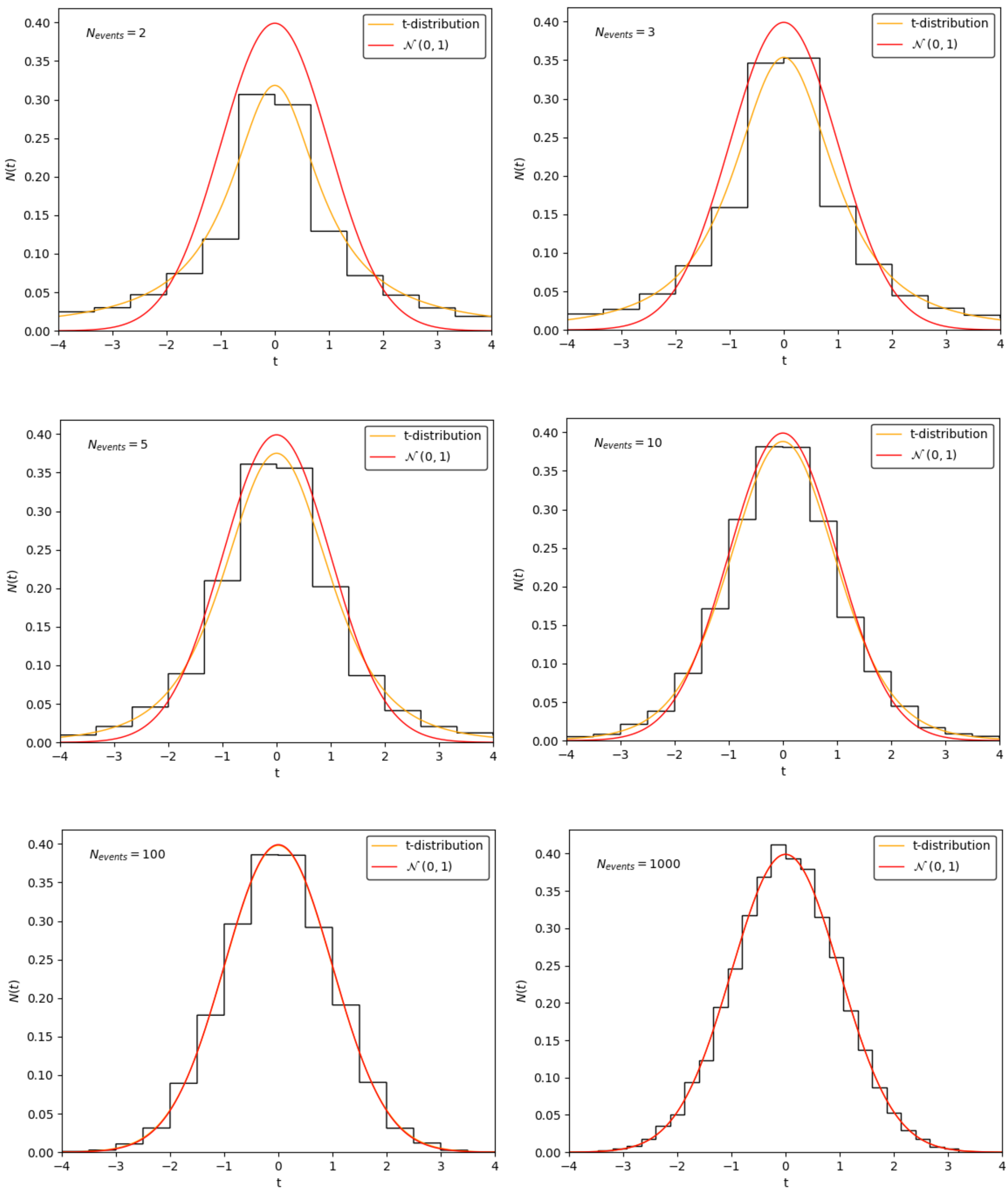


Figure 6: Histograms of the variable t from 10000 Monte Carlo experiments using $P_\mu = 0.5$ compared with the theoretical pdfs. Each histogram has been performed with different number of events.

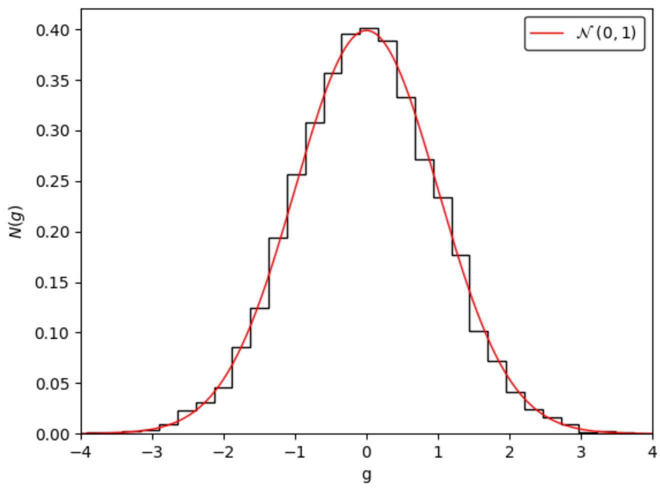


Figure 7: Histogram of the variable g from 10000 Monte Carlo experiments with 10000 measurements per experiment using $P_\mu = 0.5$ compared with the theoretical pdf.

Exercise 6

We are going to show that we can actually access to the same information using an histogram than using the basic statistical momentums of the distribution. First, we are going to see that we can estimate a pdf from a histogram. As we know the x axis of an histogram is divided into N_{bins} subintervals (bins) of width h. We denote n_i as the number of occurrences of x in the subinterval i, i.e. the number of entries in the bin. We define N as the total number of entries in the histogram $N = \sum_{i=1}^{N_{bins}} n_i$. Therefore

$$\begin{aligned} n_i &= NP(x_i - h/2 < x < x_i + h/2) \\ &= N \int_{x_i - h/2}^{x_i + h/2} dx f(x) \approx N f(x_i) h \end{aligned}$$

where we have assumed the limit of zero bin width. Then

$$f(x_i) \approx \frac{n_i}{Nh} \tag{5}$$

The pdf associated to the histogram is a multinomial distribution. However, in the limit that the number of random events N is very large, the expected pdf converges to the one that we wanted to estimate, i.e. $\frac{d\Gamma}{d\cos\theta}$ (Fig. 8).

On the other hand, the pdf associated to the number of entries per bin (content of a single bin) is a Poisson distribution. This is because we are considering the probability of finding exactly n_i events in a given interval, taking into account that the events occur independently at a constant rate. Note that it does not follow a binomial distribution precisely because $n_i p_i = \mu_i$ remains constant when $n_i \to \infty$ and $p \to 0$. In the limit that the number of random events N is very large, this distribution tends to a Normal distribution $\mathcal{N}(\mu_i, \sqrt{\mu_i})$. In order to prove this behaviour, we have performed several Monte Carlo experiments with the same bins and we have computed n_i per each experiment. Then we have focused in one single bin and we have plotted the values of this one obtained per each experiment (Fig. 9).

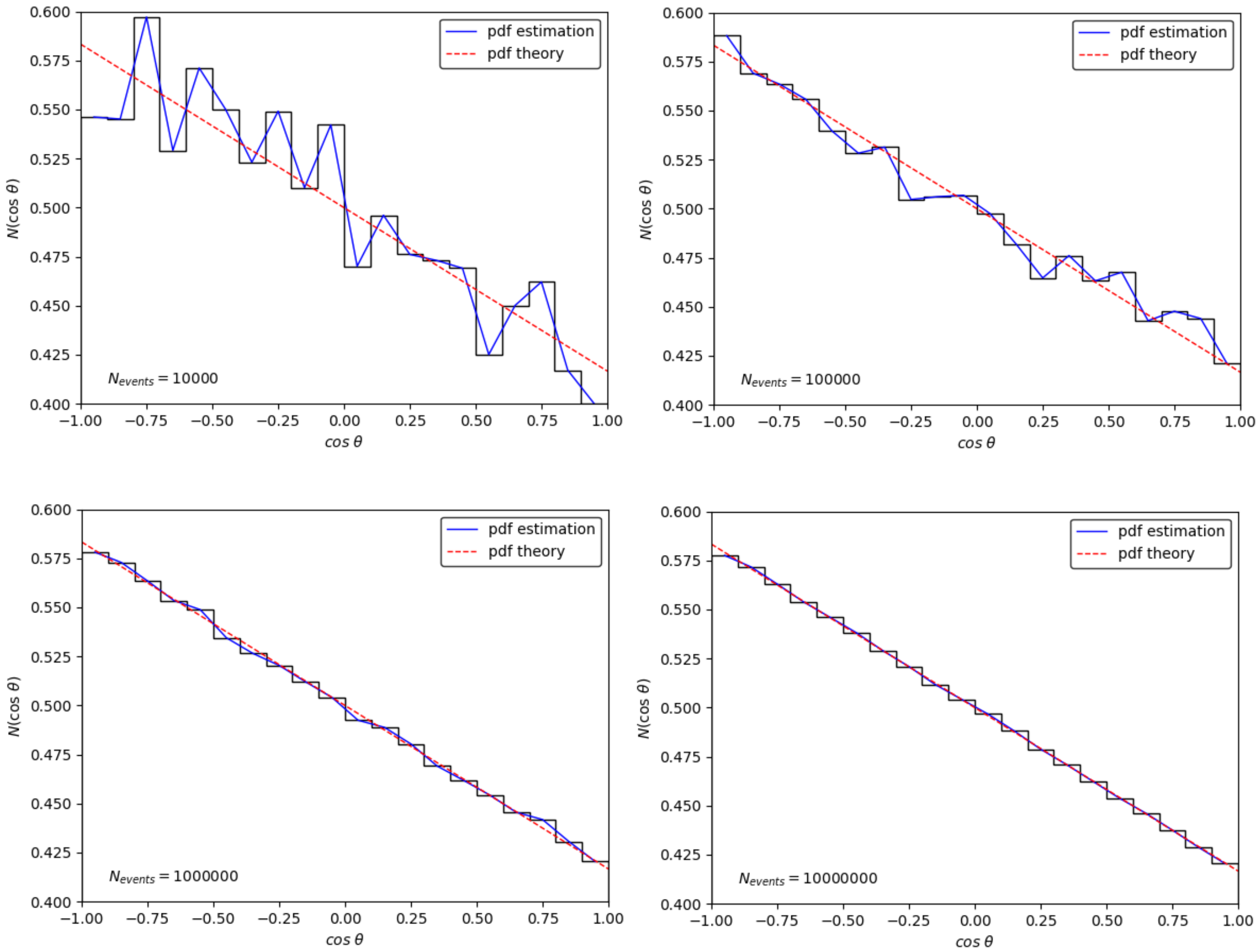


Figure 8: Histograms of the variable $\cos\theta$ from a Monte Carlo experiment using $P_\mu = 0.5$ and comparison between the estimated pdfs and the theoretical ones. Each histogram has been performed with different number of events.

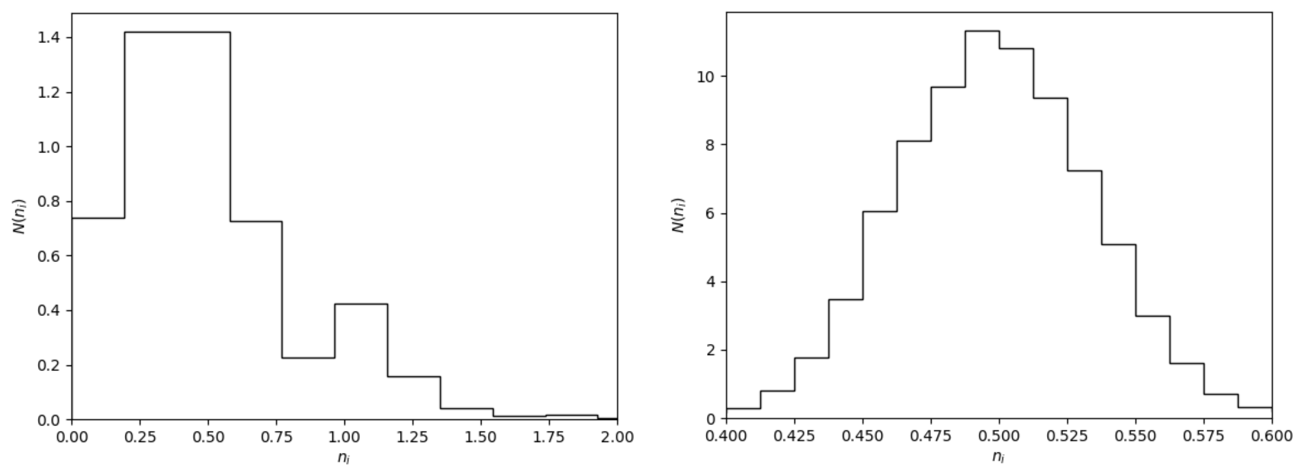


Figure 9: Histograms of the different values of a single bin n_i. We have performed 10000 Monte Carlo experiments using $P_\mu = 0.5$. From a first histogram of 50 bins we have focused on the bin number 25, which has a nominal parameter (mean) of $\mu_i \approx 0.5$. On the left, we have used 100 measurements and we can observe a Poisson distribution. On the right, we have used 10000 measurements and we can observe a Normal distribution.

Pearson's χ^2 test

We are going to perform the Pearson's χ^2 test which is a common goodness-of-fit test that can be applied to the distribution of a variable x. Given a histogram of the observed data x, we want to construct a statistic which reflects the level of agreement between the observed and the expected histogram. This statistic is given by the χ^2 variable

$$\chi^2 = \sum_{i=1}^{N_{bins}} \frac{(n_i - \mu_i)^2}{\sigma_i^2} \tag{6}$$

where n_i is the observed number of entries in the bin i, and μ_i and σ_i^2 are the means and variances of the expected histogram. The χ^2 variable resembles a normalized sum of squared deviations between the observed and the theoretical frequencies.

Given the assumptions of independent random variables (observations), simple random sample, sample sufficiently large and adequate cell counts ($n_i \geq 5$), then one can show that the statistic (Eq. 6) will follow a χ^2 distribution for N_{bins} degrees of freedom. This holds regardless of the distribution of the variable x.

A priori we do not know μ_i nor σ_i. Nevertheless, as the total number of entries N is fixed and the content of the bins follows a Gaussian distribution, we know that $\mu_i = Np_i$ and $\sigma_i^2 = Np_i$, with probabilities $p_i = \mu_i/N$. Then, we can reconstruct the χ^2 statistic as

$$\chi^2 = \sum_{i=1}^{N_{bins}} \frac{(n_i - Np_i)^2}{Np_i} \tag{7}$$

and we can observe that it follows a χ^2 distribution for $N_{bins} - 1$ degrees of freedom.

The procedure is to perform several experiments (Monte Carlos) with the same number of measurements and P_μ, and generate the histogram of the observations per each experiment in order to obtain n_i. The tricky part here is to compute μ_i. We are going to do it using the value of the original pdf formula taking x as the central value in the bin i and then multiplying it by the integral over the histogram. Once this is done, we calculate the χ^2 per each experiment using Eq. 7. Finally, we gather all the χ^2 variables and plot them in a histogram (Fig. 10).

As we can observe (Fig. 10), when the number of bins (degrees of freedom) $N_{bins} \to \infty$ the χ^2 distribution converges to a normal distribution $\mathcal{N}(N_{bins}, \sqrt{2N_{bins}})$. In addition, we can notice that when the number of bins of the histogram is increased, some bins exhibit discrepancies. This is because of the expected statistical fluctuations.

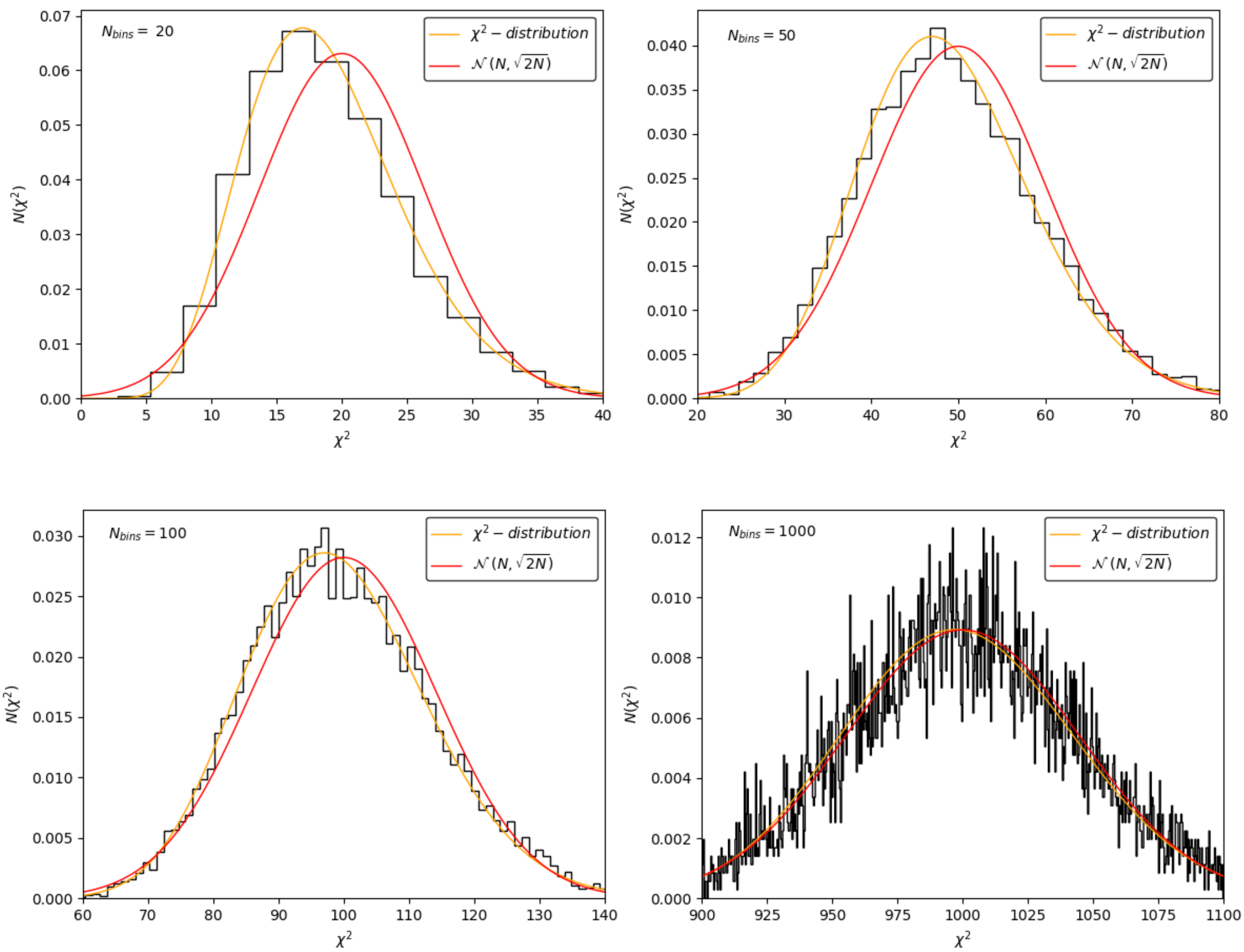


Figure 10: Histograms of the variable χ^2 from 10000 Monte Carlo experiments with 10000 measurements per experiment using $P_\mu = 0.5$ compared with the theoretical pdfs. Each histogram has been performed with different number of bins.

Exercise 7

We are going to construct the conditional probability $P(histogram|P_\mu)$ of obtaining the entries in the obtained histogram for a given value P_μ. This is in fact the so-called likelihood function $L(P_\mu)$ defined as

$$L(\theta) = \prod_i^n f(x_i, \theta) \tag{8}$$

where $f(x_i, \theta)$ is the pdf of the random variable x_i and θ are the parameter or parameters unknown, which in our case is P_μ. Note that x_i is fixed, i.e. the experiment is over.

The likelihood function is used especially in methods of estimating a parameter from a set of statistics, since, as we will see, a parameter value close to the true value should yield a high probability for the measurements obtained. In order to construct $L(P_\mu)$, we proceed as we did in exercise 6, so we generate N random events according to the pdf $\frac{d\Gamma}{d\cos\theta}$ and we fill the obtained values in a histogram. We compute this single Monte Carlo experiment with a certain value P_μ, which will be our "true value". Note that we are generating our measurements from a certain parameter P_μ. This implies that at the end of the computation, out estimated $\hat{P}_\mu$ should be in fact this P_μ. However, it is important to highlight that in a general case we would not know this "true value" and that is why we apply this method.

As we saw in Fig. 9, for N large, the content of the bins n_i follows a Normal distribution $\mathcal{N}(\mu_i, \sqrt{\mu_i})$. Then what we have is a set of N_{bins} independent Gaussian random variables n_i, $i = 1, ..., N_{bins}$. This in terms of the likelihood (Eq. 8) will correspond to the product of N_{bins} Gaussians:

$$L(P_\mu) = \prod_i^{N_{bins}} \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(\frac{-(n_i - \lambda_i(x_i; P_\mu))^2}{2\sigma_i^2}\right) \tag{9}$$

where n_i is the number of entries in the bin i, σ^2 is the variance of the number of entries in the bin i and λ_i is the number of entries predicted $\lambda_i = E[n_i]$. Taking into account that the content of each bin is Gaussian distributed, the variance is therefore equal to the mean $\sigma_i^2 = E[n_i]$, so Eq. 9 becomes

$$L(P_\mu) = \prod_i^{N_{bins}} \frac{1}{\sqrt{2\pi\lambda_i(P_\mu)}} \exp\left(\frac{-(n_i - \lambda_i(P_\mu))^2}{2\lambda_i(P_\mu)}\right) \tag{10}$$

Now, in order to compute $L(P_\mu)$, first we should obtain $\lambda_i(P_\mu)$. As we did in exercise 6, we are going to acquire them from the original pdf formula. On the other hand, n_i are the bin counts from our single histogram. Considering this, we do our calculations for a range of values of P_μ and we obtain the Fig. 11.

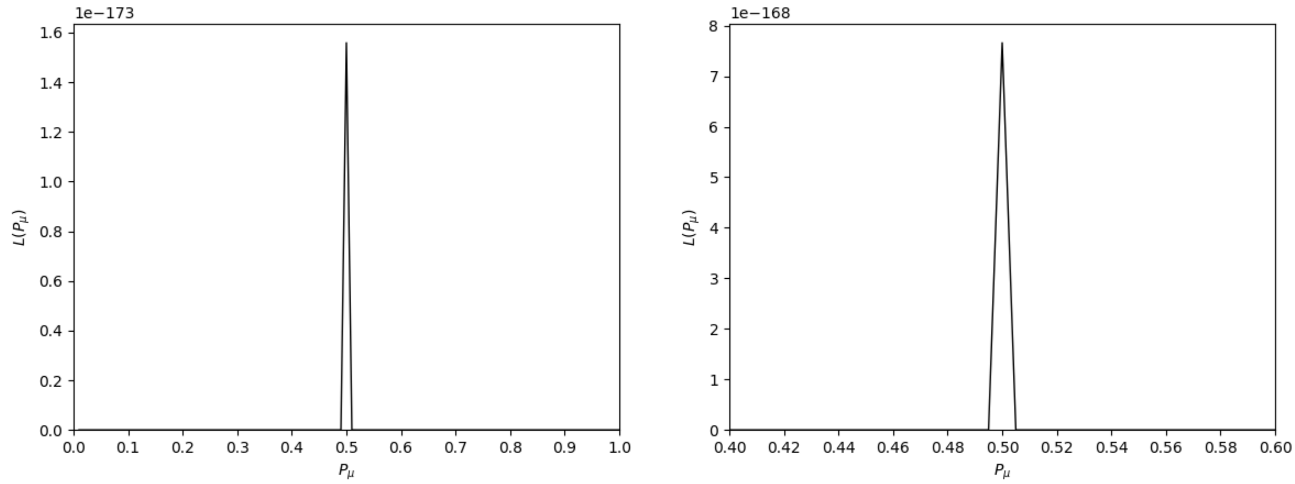


Figure 11: Likelihood $L(P_\mu)$ as a function of the parameter P_μ. The histogram used has been performed from a single Monte Carlo with 10000000 measurements, 50 bins and the "true value" $P_\mu = 0.5$. On the left, we have used a P_μ step width of 0.01. On the right, we have used a step width of 0.005.

The parameter estimator $\hat{P}_\mu$ is the most probable value of the likelihood $L(P_\mu)$, i.e. the most probable value to generate the histogram. The result obtained is $\hat{P}_\mu = 0.5$, which coincides with our "true value" as we expected. The precision of this result is given by the step width of P_μ used, which in our case is 0.005.

As mentioned, the likelihood function is used in methods of parameter estimation. The way we have proceeded is the so-called method of maximum likelihood. In our case we have computed the entire likelihood because we wanted to plot it. However, in general one takes the logarithm of the likelihood, drops the additive terms that do not depend on the parameters and maximize the resultant log-likelihood function. On the other hand, the other important method is the method of least squares. This one consists in finding the parameters that minimize the quantity χ^2. Both methods are related with the likelihood. As we can observe, if we perform the method of LS, we obtain the same result (Fig. 12).

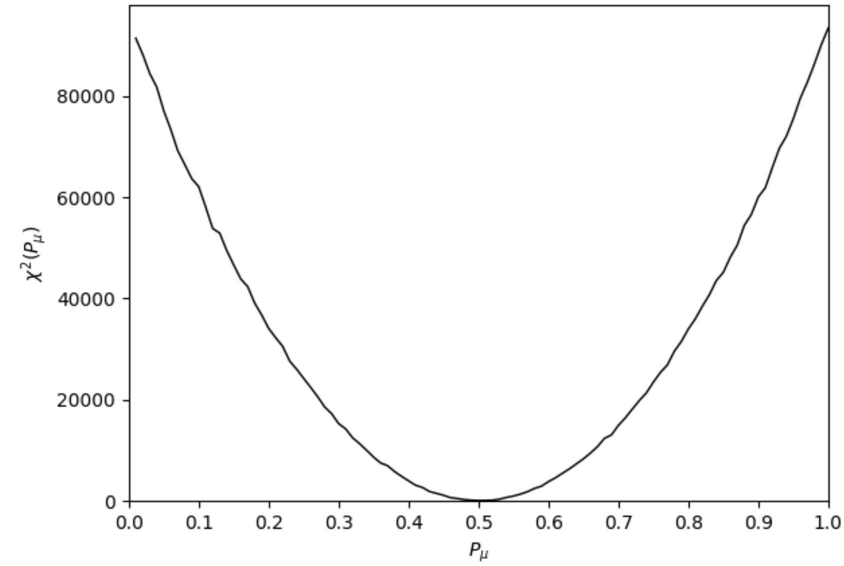


Figure 12: $\chi^2(P_\mu)$ as a function of the parameter P_μ. The histogram used has been performed from a single Monte Carlo with 10000000 measurements, 50 bins and the "true value" $P_\mu = 0.5$.