Ćwiczenie 1

Hipoteza Statystyczna

1. Cel ćwiczenia

Celem ćwiczenia jest zaznajomienie się z metodami statystycznymi pozwalającymi na weryfikację hipotezy dotyczącej rodzaju lub parametrów rozkładu prawdopodobieństwa badanej populacji na podstawie pobranej próbki losowej.

2. Popularne rozkłady prawdopodobieństwa zmiennej ciągłej.

2.1 Rozkład normalny

Rozkład normalny, zwany również rozkładem Gaussa (Gaussa-Laplace'a) jest najczęściej występującym w literaturze. Każda zmienna losowa, która jest sumą wielu czynników, niezależnie od rozkładu prawdopodobieństwa tych czynników, będzie zmienną z rozkładu normalnego. Jest to wynikiem działania centralnego twierdzenia granicznego.

Funkcja gęstości prawdopodobieństwa zmiennej x z rozkładu normalnego o wartości oczekiwanej μ oraz odchyleniu standardowym σ jest opisana wzorem

$$N(x;\ \mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right). \tag{1}$$

Jeśli $\mu = 0$ oraz $\sigma = 1$, to rozkład nazywa się standaryzowanym rozkładem normalnym. Jeśli zmienna x pochodzi z rozkładu normalnego $N(x;\ \mu,\sigma)$, to zmienna $\tilde{x} = \frac{x-\mu}{\sigma}$ pochodzi z rozkładu $N(x;\ 0,1)$.

W statystyce niezwykle ważne jest pojęcie dystrubuanty. Jest to prawdopodobieństwo P, że zmienna losowa X ma wartości mniejsze bądź równe x. Dla rozkładu normalnego dystrybuanta wyraża się wzorem

$$\mathrm{P}(X \le x;\ \mu,\sigma)\ = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(t-\mu)^2}{2\sigma^2}\right)\mathrm{d}t. \tag{2}$$

Wzór (2) nie ma postaci analitycznej. W zastosowaniu, całkę oblicza się numerycznie lub korzysta z wartości tablicowych. Wartości dystrybuanty (2) są stablicowane dla rozkładu standaryzowanego. Wartość dystrubuanty dla dowolnego rozkładu normalnego można otrzymać z relacji

$$\mathrm{P}(X \le x;\ \mu,\sigma) = \Phi\left(\frac{x-\mu}{\sigma}\right), \tag{3}$$

gdzie $\Phi(x)$ jest dystrybuantą standaryzowanego rozkładu Gaussa.

Istnieje funkcja, która jest predefiniowana w bibliotekach matematycznych języków programowania, pozwalająca na obliczenie dystrubuanty Φ. Jest to tzw. funkcja błędu oznaczana skrótowo jako erf. Związek między dystrybuantą Φ a funkcją błędu jest następujący:

$$\Phi(x) = \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{x}{\sqrt{2}}\right)\right). \tag{4}$$

2.2 Rozkład χ^2

Niech zmienna losowa u będzie sumą niezależnych zmiennych losowych x_i z tego samego rozkładu normalnego o wartości oczekiwanej μ oraz odchyleniu standardowym σ:

$$\mathrm{u} = \sum_{i=1}^{n} \left(\frac{x_i - \mu}{\sigma}\right)^2. \tag{5}$$

Rozkład zmiennej u jest rozkładem χ^2 o n stopniach swobody:

$$\chi_\mathrm{n}(u) = \frac{1}{\left(\sqrt{2}\right)^n \Gamma\left(\frac{n}{2}\right)} u^{\frac{n}{2}-1} \exp\left(-\frac{u}{2}\right). \tag{6}$$

2.3 Rozkład Studenta

Zauważmy, że zmienna u opisana wzorem (5) wymaga znajomości odchylenia standardowego rozkładu normalnego, z którego pochodzą próbki x_i. Jest ona jednak często niedostępna. W celu uniknięcia tego problemu można rozważyć zmienną t postaci:

$$\mathrm{t} = \frac{\bar{x} - \mu}{s_{\bar{x}}}, \tag{7}$$

gdzie $\bar{x}$ – estymator wartości oczekiwanej, $s_{\bar{x}}$ – estymator odchylenia standardowego średniej. Zmienna t ma rozkład w postaci rozkładu Studenta o n stopniach swobody:

$$\mathrm{S_n}(t) = \frac{\Gamma\left(\frac{1}{2}(n+1)\right)}{\sqrt{\pi n}\,\Gamma\left(\frac{n}{2}\right)} \left(1 + \frac{t^2}{n}\right)^{-\frac{1}{2}(n+1)} \tag{8}$$

2.4 Rozkład Snedecora-Fishera

Rozważmy iloraz dwóch zmiennych u_1 oraz u_2, każda o rozkładzie χ^2 o odpowiednio n i m stopniach swobody:

$$F = \frac{u_1}{n} \Big/ \frac{u_2}{m}. \tag{9}$$

Rozkład zmiennej F jest rozkładem Snedeckora-Fishera o (n, m) stopniach swobody:

$$F_{n,m}(F) = \frac{\Gamma\left(\frac{1}{2}(n+m)\right) n^{\frac{n}{2}} m^{\frac{m}{2}} F^{\frac{n}{2}-1}}{\Gamma\left(\frac{n}{2}\right)\Gamma\left(\frac{m}{2}\right)(m+nF)^{\frac{1}{2}(n+m)}}. \tag{10}$$

Statystyki Snedecora-Fishera można użyć do porównania odchyleń standardowych w dwóch próbkach wylosowanych z rozkładu normalnego. W tym celu używamy zmienną Fishera:

$$F = \frac{s_x^2}{\sigma_x^2} \Big/ \frac{s_y^2}{\sigma_y^2}. \tag{11}$$

Zmienna ta podlega rozkładowi Snedecora-Fishera o $(n-1, m-1)$ stopniach swobody. Najczęstsze użycie zmiennej (11) dotyczy testu prawdziwości hipotezy, że próbki x oraz y pochodzą z rozkładów normalnych o równych odchyleniach standardowych.

3. Wykonanie ćwiczenia

3.1. Pomiar

Eksperyment dotyczy 100-krotnego pomiaru dwukrotnego okresu drgań wahadła matematycznego. Wahadło wprowadza się w ruch jednokrotnie poprzez wychylenie ciężarka od położenia równowagi. Wychylenie powinno być małe ($< 7°$), aby na statystykę pomiaru nie wpływała zależność okresu drgań od kąta wychylenia. Po wychyleniu wahadła stoper należy włączyć, kiedy ciężarek znajduje się w maksymalnym wychyleniu. Za każdym razem, kiedy wahadło dokona dwóch pełnych wychyleń, należy zapisać czas, w którym te wychylenia nastąpiły.

3.2. Histogram i Rozkład Normalny

Okres drgań dla każdego pojedynczego pomiaru uzyskanego zgodnie z punktem 3.1 należy otrzymać poprzez podzielenie zmierzonego czasu przez 2. Aby otrzymać histogram okresu drgań wahadła należy podzielić przedział, w którym zawierają się zmierzone okresy drgań na k przedziałów. Szerokość h przedziału (słupka histogramu) otrzymujemy ze wzoru

$$h = \frac{T_{max} - T_{min}}{k}, \tag{12}$$

gdzie T_{max} oraz T_{min} to odpowiednio maksymalny i minimalny okres drgań wynikający z pomiarów.

Liczbę przedziałów histogramu można dobrać „na oko", jednak istnieją pewne użyteczne kryteria, pomagające w określeniu tego parametru. Jednym z nich jest zasada Freedmana-Diaconisa. Założeniem tej metody jest minimalizacja różnicy pola powierzchni między histogramem a teoretycznym rozkładem prawdopodobieństwa. Ogólne równanie na szerokość przedziału histogramu:

$$h = 2\frac{\mathrm{IQR}(x)}{\sqrt[3]{n}}, \tag{13}$$

gdzie x – populacja dla której wykonujemy histogram, IQR – rozstęp ćwiartkowy (ang. *interquartile range*), n – liczba próbek. Aby otrzymać rozstęp ćwiartkowy należy obliczyć pierwszy i trzeci kwartyl i odjąć pierwszy od trzeciego. Pierwszy kwartyl to mediana z połowy najmniejszych wartości populacji, a kwartyl trzeci to mediana z połowy największych wartości populacji.

Wykorzystajmy program Matlab do narysowania histogramu okresu drgań wahadła matematycznego zmierzonego w punkcie 3.1. Stwórzmy nowy skrypt o nazwie `plot_histogram.m`. Rozszerzenie `*.m` jest charakterystyczne dla plików programu Matlab. Załóżmy, że dane są zapisane w jednej kolumnie w pliku tekstowym `dane.txt`. Pierwszą czynnością jest załadowanie danych do programu. Import danych dokonuje się funkcją `load('nazwa_pliku')`. Ważne, żeby plik z danymi znajdował się w tej samej ścieżce co skrypt programu. Wynik działania funkcji należy przypisać do zmiennej np. o nazwie `dane`. Stwórzmy dodatkową zmienną przechowującą rozmiar wektora danych, co można otrzymać komendą `n=length(dane)`.

Po wczytaniu danych możemy obliczyć parametry histogramu. Musimy obliczyć szerokość h pojedynczego binu i liczbę przedziałów histogramu. Szerokość obliczamy ze wzoru (13) wpisując następującą instrukcję do skryptu: `h=2*iqr(dane)/n^(1/3)`. Funkcja `iqr` wywołana dla wektora danych wejściowych zwraca rozstęp ćwiartkowy. Liczbę k binów obliczymy przekształcając

wzór (12). Wielkości T_{max} oraz T_{min} obliczymy wykorzystując odpowiednio funkcje `max` oraz `min` wywołane na wektorze `dane`. Obliczona wartość k może nie być liczbą całkowitą, dlatego dodatkowo należy ją sprowadzić do najbliższej liczby całkowitej poprzez zaokrąglenie np. funkcją `round`.

Możemy przystąpić do narysowania histogramu. Służy do tego funkcja `hist`, przyjmująca dwa argumenty: pierwszy argument to wektor danych, a drugi liczba binów hostogramu. Funkcja zwraca dwie zmienne: wektor liczby zliczeń w danym binie oraz środek binu. Po znalezieniu tych wartości wykres histogramu w postaci słupków można narysować funkcją `bar` przyjmującą jako pierwszy argument wektor centrów binów, a drugi argument to wektor zliczeń w danym binie. Histogram w tej postaci nie będzie jednak poprawnie znormalizowany. Aby dokonać normalizacji musimy podzielić liczbę zliczeń w danym binie przez całkowitą liczbę populacji (zmienna `n`) oraz dodatkowo podzielić przez szerokość pojedynczego binu. Jeśli wektor pozycji centrów binu oznaczymy jako `h_bin`, to szerokość binu wynosi `w_bin=h_bin(2)-h_bin(1)`. Dopiero po dokonaniu tej normalizacji rysujemy wykres słupkowy funkcją `bar`.

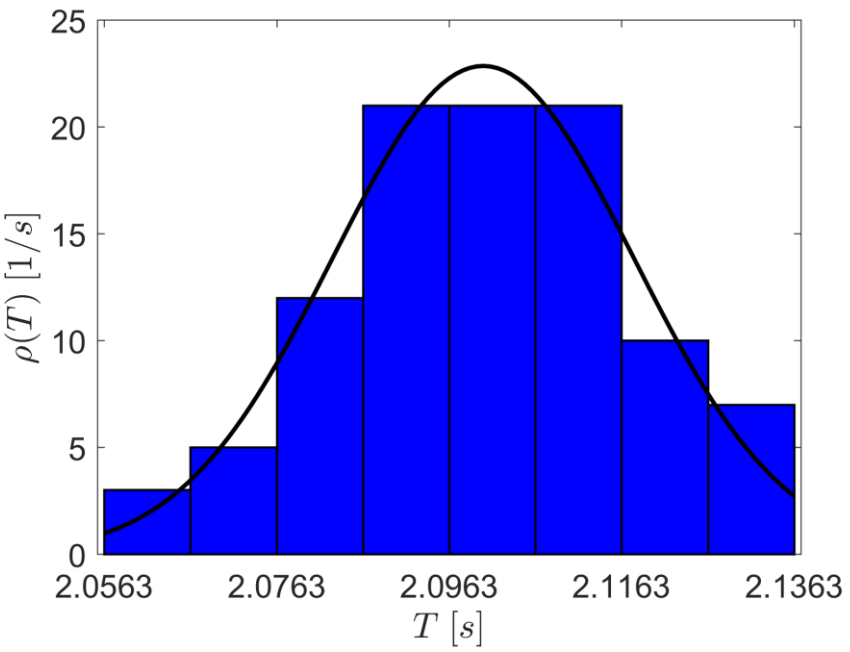
Spodziewamy się, że otrzymany histogram będzie przypominał rozkład normalny. Wynika to z randomowego sposobu pomiaru okresu drgań. Na mierzony czas wpływa wiele czynników np. szybkość reakcji palca przy naciskaniu przycisku zatrzymania czasu czy przypadkowego wyboru momentu, kiedy wahadło wykonuje dwa pełne okresy ruchu. Zgodnie z centralnym twierdzeniem granicznym suma takich randomowych czynników, niezależnie od rozkładu poszczególnych czynników, ma zawsze rozkład normalny. Warto sprawdzić czy teoretyczna krzywa dzwonowa (krzywa Gaussa) dobrze dopasowuje się do naszego histogramu. Aby ją narysować na tle histogramu należy najpierw obliczyć dwa parametry rozkładu normalnego: wartość oczekiwaną oraz odchylenie standardowe. W tym celu posłużymy się estymatorem wartości oczekiwanej w postaci średniej arytmetycznej oraz estymatorem odchylenia standardowego. Wzory są następujące:

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i, \qquad \hat{\sigma} = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \hat{\mu})^2}{n-1}}, \tag{14}$$

gdzie x_i to pojedynczy pomiar. Wygodnie posłużyć się funkcjami Matlaba, gdzie $\hat{\mu}$ można obliczyć funkcją `mean`, natomiast $\hat{\sigma}$ otrzymamy dzięki funkcji `std`. **Uwaga! W niektórych bibliotekach funkcja `std` może mieć inną implementację, gdzie zamiast n-1 pojawia się tylko n. Dokładną definicję używanej funkcji należy zawsze sprawdzić przed wykonaniem obliczeń.** Wykorzystując parametry obliczone wzorem (14) na naszym zbiorze danych obliczamy wartość funkcji Gaussa dzięki wzorowi (1), wstawiając $\hat{\mu}$ oraz $\hat{\sigma}$ w miejsce μ oraz σ. Aby pokazać ciągłość tej funkcji należy obliczyć ją dla punktów z przedziału $[T_{min}, T_{max}]$ o większej gęstości niż liczba binów np. dla 100 równoodległych punktów. Zdefiniowanie wektora zmiennej niezależnej dla funkcji Gaussa można zrobić komendą: `x=T_min:(T_max-T_min)/100:T_max.` Wektor zmiennej zależnej uzyskamy stosując bezpośrednio wzór (1): `y=1/sqrt(2*pi)/std_est*exp(-(x-mi_est).^2/2/std_est^2).` W tym wypadku zmienne `std_est` oraz `mi_est` to odpowiednio $\hat{\mu}$ oraz $\hat{\sigma}$. Narysowanie funkcji 1D odbywa się komenda `plot`. Kod programu obliczającego parametry histogramu oraz rysującego rozkład normalny na podstawie obliczonych parametrów populacji przezentuje Listing 1. Wykres prezentujący histogram wraz z krzywą Gaussa prezentuje Rysunek 1. Oprócz wspomnianych funkcji i komend Matlaba, na Listingu 1. Pojawiają się dodatkowe komendy i opcje służące poprawieniu estetyki generowanego wykresu. Dokładny opis użytych opcji najlepiej sprawdzić w dokumentacji programu Matlab, jako że wykracza to poza materiał ćwiczenia.

```
clear
clc

dane=load('dane.txt');
n=length(dane);
h=2*iqr(dane)/n^(1/3);
T_max=max(dane);
T_min=min(dane);
k=round((T_max-T_min)/h);

[n_bin, h_bin]=hist(dane, k);
n_bin=n_bin/n/(h_bin(2)-h_bin(1));

mi_est=mean(dane);
std_est=std(dane);
x=T_min:(T_max-T_min)/100:T_max;
y=1/sqrt(2*pi)/std_est*exp(-(x-mi_est).^2/2/std_est^2);

figure
hold on
bar(h_bin, n_bin, 1, ...
    'FaceColor', 'b', 'EdgeColor', 'k', 'LineWidth', 1);
plot(x, y, '-k', 'LineWidth', 2);
xlabel('$T \; [s]$', 'Interpreter', 'Latex', 'FontSize', 15);
ylabel('$\rho (T)\; [1/s]$', 'Interpreter', 'Latex', 'FontSize', 15);
set(gca, 'Box', 'on', 'Fontsize', 15, 'Xtick', [T_min, ...
    (T_max-T_min)/4+T_min,(T_max-T_min)/2+T_min 3*(T_max-T_min)/4+T_min, T_max]);
xlim([T_min-(T_max-T_min)/100, T_max+(T_max-T_min)/100]);

l=legend({'histogram', 'Gauss\_est', 'Gauss\_fit'});
set(l, 'FontSize', 10);
```

Listing 1. Kod źródłowy skryptu `plot_histogram.m` obliczający parametry histogramu dla danych znajdujących się w pliku `dane.txt`.



Rysunek 1. Histogram dla przykładowych 100 pomiarów okresu drgań wahadła matematycznego wraz z dorysowaną krzywą Gaussa.

3.3 Dopasowanie krzywej Gaussa do histogramu.

W punkcie 3.2 używaliśmy estymatorów wartości oczekiwanej oraz odchylenia standardowego aby znaleźć parametry rozkładu normalnego. Dopasowanie krzywej Gaussa do histogramu można przeprowadzić na drugi sposób, gdzie parametry krzywej Gaussa zostaną dobrane tak, aby różnice między krzywą modelową a histogramem były zminimalizowane np. używając metody najmniejszych kwadratów.

Zauważmy, że utworzywszy histogram dostajemy wektor centrów binów oraz wektor wysokości binów. Zakładając, że wektor centrów jest zmienną niezależną x, a wektor wysokości binów jest zmienną zależną y można dopasować do danych funkcję $y(x)$ w postaci funkcji Gaussa (wzór (1)), gdzie μ oraz σ są parametrami modelu.

W programie Matlab istnieje funkcja pozwalająca na dopasowanie dowolnego modelu krzywej czy powierzchni 2D do danych doświadczalnych dzięki funkcji `fit`. Istnieje też sposób dopasowania dowolnej funkcji n-wymiarowej przy użyciu funkcji `fmincon`, ale jej złożoność wymagałaby odrębnego ćwiczenia.

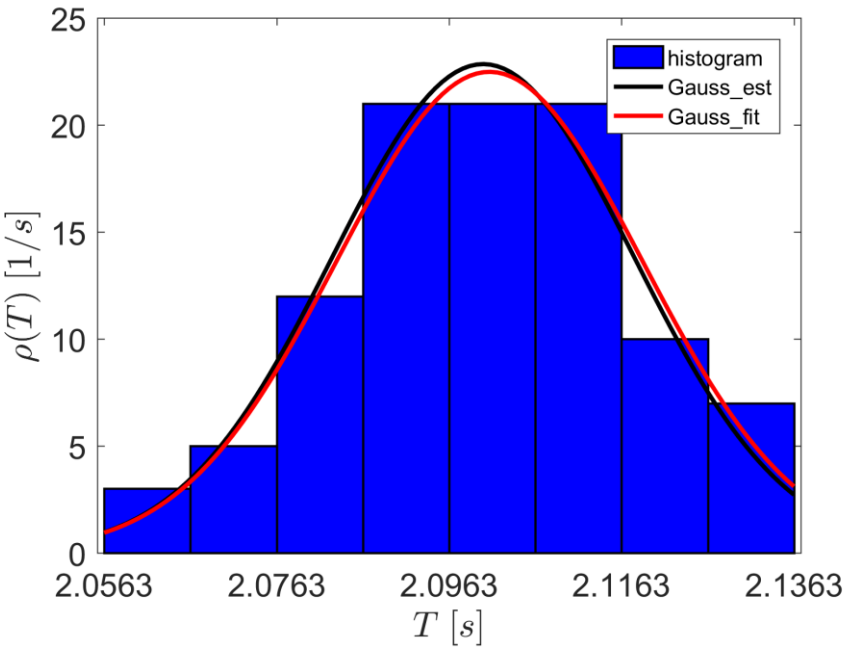
Funkcja `fit` wymaga przekazania do funkcji wektora zmiennej niezależnej, wektora zmiennej zależnej oraz zdefiniowania modelu, który będzie dopasowany. Istnieją również dodatkowe opcje pozwalające na zadanie parametrów startowych dopasowania, nałożenia więzów na parametry czy wybór predefiniowanych modeli. Skorzystamy tylko z opcji `StartPoint`, która pozwala zadać parametry początkowe algorytmu.

Zanim przystąpimy do wywołania funkcji `fit` należy zdefiniować model oraz opcje tego modelu. Model najlepiej zdefiniować poprzez użycie tzw. funkcji anonimowej. Składnia przygotowania takiej funkcji w postaci krzywej Gaussa w Matlabie wygląd anastępująco: `my_fit_fun=@(sigma, mi, x) 1/sqrt(2*pi)/sigma*exp(-(x-mi).^2/2/sigma^2);`. Nazwa funkcji to `my_fit_fun`, ale można nazwać ją dowolnie. Po znaku przypisania pojawia się definicja parametrów modelu, odpowiadających zmiennym σ, μ oraz zmiennej niezależnej x. Parametry umieszczone są w nawiasie po znaku @. Następnie pojawia się wzór modelu, w tym wypadku jest to wzór znormalizowanej krzywej Gaussa. Jest to koniec funkcji anonimowej. W programie można wywołać ją dla dowolnych wartości parametrów. W nowej linii kodu podamy parametry modelu. Należy wskazać który parametr będzie dopasowany, a który zostanie podany przez użytkownika. W naszym przypadku parametry `sigma` oraz `mi` będą dopasowane. Wywołujemy funkcję `fittype`, która wskaże programowi który model zostanie dopasowany i z jakimi parametrami. Wywołanie tej funkcji jest następujące: `my_fit_def=fittype(my_fit_fun, 'coefficients', {'sigma', 'mi'}, 'independent', 'x');`. Do funkcji `fittype` najpierw przekazaliśmy nazwę modelu, czyli `my_fit_fun` zdefiniowany wcześniej funkcją anonimową. Następnie wskazujemy które parametry są zmiennymi do dopasowania etykietą `coefficients`, oraz który parametr jest zmienną niezależną etykietą `independent`. Zmienna `my_fit_def` przechowuje informację o opcjach modelu. Teraz można już wywołać funkcję dopasowującą `fit`. Składnia wywołania wygląda następująco: `my_fit=fit(h_bin', n_bin', my_fit_def, 'StartPoint', [0.1, 2]);`. Zmienna `my_fit` przechowuje parametry dopasowanego modelu. Aby pobrać wartość parametru `sigma` należy wpisać instrukcję `my_fit.sigma`. Analogicznie dla parametru `mi` instrukcja to `my_fit.mi`. Do funkcji `fit` przekazany został wektor położeń centrów binów (`h_bin`) oraz wektor wysokości binu po normalizacji (`n_bin`). Warto zauważyć, że po nazwie zmiennej pojawia się apostrof. Jest on potrzebny, ponieważ zmienne są w postaci wektorów wierszowych, natomiast funkcja dopasowująca wymaga wektorów kolumnowych. Apostrof jest operatorem wykonującym operację transpozycji.

Uaktualnioną wersję programu wraz z algorytmem dopasowującą krzywą Gaussa oraz rysującą ją na wspólnym wykresie prezentuje Listing 2. Uaktualniony wykres o dopasowaną krzywą Gaussa prezentuje Rysunek 2.

```
clear
clc

dane=load('dane.txt');
n=length(dane);
h=2*iqr(dane)/n^(1/3);
T_max=max(dane);
T_min=min(dane);
k=round((T_max-T_min)/h);

[n_bin, h_bin]=hist(dane, k);
n_bin=n_bin/n/(h_bin(2)-h_bin(1));

mi_est=mean(dane);
std_est=std(dane);
x=T_min:(T_max-T_min)/100:T_max;
y=1/sqrt(2*pi)/std_est*exp(-(x-mi_est).^2/2/std_est^2);

my_fit_fun=@(sigma, mi, x) 1/sqrt(2*pi)/sigma*exp(-(x-mi).^2/2/sigma^2);
my_fit_def=fittype(my_fit_fun, 'coefficients', {'sigma', 'mi'}, 'independent', 'x');
my_fit=fit(h_bin', n_bin', my_fit_def, 'StartPoint', [0.1, 2]);

figure
hold on
bar(h_bin, n_bin, 1, ...
    'FaceColor', 'b', 'EdgeColor', 'k', 'LineWidth', 1);
plot(x, y, '-k', 'LineWidth', 2);
plot(x, my_fit_fun(my_fit.sigma, my_fit.mi, x), '-r', 'LineWidth', 2);
xlabel('$T \; [s]$', 'Interpreter', 'Latex', 'FontSize', 15);
ylabel('$\rho (T)\; [1/s]$', 'Interpreter', 'Latex', 'FontSize', 15);
set(gca, 'Box', 'on', 'Fontsize', 15, 'Xtick', [T_min, ...
    (T_max-T_min)/4+T_min,(T_max-T_min)/2+T_min 3*(T_max-T_min)/4+T_min, T_max]);
xlim([T_min-(T_max-T_min)/100, T_max+(T_max-T_min)/100]);

l=legend({'histogram', 'Gauss\_est', 'Gauss\_fit'});
set(l, 'FontSize', 10);
```

Listing 2. Kod programu obliczającego parametry histogramu oraz dopasowującego krzywą Gaussa.



Rysunek 2. Histogram z krzywą Gaussa narysowaną w oparciu o estymowane parametry modelu (*Gauss_est*) oraz dopasowaną metoda najmniejszych kwadratów (*Gauss_fit*).

3.4 Test Studenta wartości oczekiwanej.

W punkcie 3.3 otrzymaliśmy krzywą Gaussa z dopasowania parametrów rozkładu normalnego do histogramu. Istnieje pytanie czy krzywa dopasowana jest statystycznie równoważna krzywej wynikającej z wartości parametrów obliczonych estymatorami. W tym celu wykonamy test Studenta dla wartości oczekiwanej.

Test Studenta polega na sprawdzeniu czy zmienna Studenta wyrażona wzorem (7), przy założeniu, że znamy wartość μ zawiera się w przedziale $\left[t\left(\frac{\alpha}{2}\right), t\left(1-\frac{\alpha}{2}\right)\right]$, gdzie $t\left(\frac{\alpha}{2}\right)$ to kwantyl rozkładu Studenta o $n-1$ stopniach swobody, dla którego prawdopodobieństwo p obliczone dla przedziału $\left[-\infty, t\left(\frac{\alpha}{2}\right)\right]$ jest równe α. Analogicznie $t\left(1-\frac{\alpha}{2}\right)$ to kwantyl, dla którego prawdopodobieństwo obliczone dla przedziału $\left[-\infty, t\left(1-\frac{\alpha}{2}\right)\right]$ jest równe $1-\frac{\alpha}{2}$. Wartość α nazywana jest poziomem istotności testu i często jest równa 0.05.

Dla danych prezentowanych na Rysunku 2. estymowana wartość oczekiwana okresu drgań wynosi $T_{est} = 2.102$ s. Analogiczna wartość z dopasowania rozkładu normalnego wyniosła $T_{fit} = 2.101$ s. Estymowana wartość odchylenia standardowego okresu drgań wyniosła $\sigma_T^{est} = 0.0175$ s. Obliczamy wartość zmiennej z rozkładu Studenta wzorem (7):

$$t = \frac{T_{est} - T_{fit}}{\frac{\sigma_T^{est}}{\sqrt{n}}} = \frac{2.102\ \text{s} - 2.101\ \text{s}}{\frac{0.0175}{\sqrt{100}}\ \text{s}} = 0.5714.$$

Wartość kwantylu rozkładu Studenta o $n-1$ stopniach swobody można odczytać z tablic statystycznych. W Matlabie istnieje funkcja `tcdf`, która oblicza prawdopodobieństwo p dla przedziału $[-\infty,\ x]$, gdzie x jest argumentem przekazywanym do funkcji. Jeśli po wywołaniu funkcji dla naszej wartości t obliczone prawdopodobieństwo $1-p$ będzie mniejsze niż $\frac{\alpha}{2}$, czyli w naszym przypadku 0.025, to znaczy, że wartość oczekiwana rozkładu populacji różni się od wartości oczekiwanej krzywej dopasowania. Dla $t = 0.5714$ i 99 stopni swobody prawdopodobieństwo Studenta wynosi $p = 0.7155$ (wywołanie w Matlabie: `tcdf(0.5714, 99)`). Wartość $1-p$ jest więc większa niż założony próg istotności ($\frac{\alpha}{2} = 0.025$). Oznacza to, że na poziome istotności równym α rozkład dopasowany oraz wyznaczony z estymatorów są sobie równoważne.

Istnieje funkcja bezpośrednio wykonująca test Studenta. Jest to funkcja `ttest`. Wywołanie dla naszych danych to: `[decision, p_score]=ttest(dane, my_fit.mi, 0.05, 'both')`. Pierwszym argumentem jest wektor danych, drugim wartość oczekiwana względem której statystykę sprawdzamy, trzecim jest poziom istotności, a ostatnim etykieta rodzaju testu. W naszym przypadku sprawdzaliśmy czy jest możliwość, że wartość oczekiwana populacji jest różna od wartości dopasowywanej, ale można sprawdzić też czy jest większa lub mniejsza. Wtedy inny jest warunek odrzucenia hipotezy, a wbudowana funkcja Matlaba uwzględnia wszystkie przypadki. Zmienna `decision` przyjmuje wartość 0 lub 1 w przypadku kiedy wartość oczekiwana jest odpowiednio równa lub różna wartości zakładanej. Zmienna `p_score` to prawdopodobieństwo, że obie krzywe normalne są statystycznie równoważne.

Uwaga! Dla populacji o liczbie próbek $n \geq 100$ rozkład Studenta sprowadza się do rozkładu normalnego, więc zamiast wyznaczać kwantyle dla rozkładu Studenta można je sprawdzać dla rozkładu normalnego.

Podobny test można wykonać dla porównania odchyleń standardowych testem Scedecora-Fishera, jednak nie będzie to tematem zajęć. W większości przypadków interesuje nas czy zmierzona wartość parametru jest zgodna z wartością tablicową, co jest przedmiotem testu Studenta.

3.5. Test χ^2 Pearsona

W poprzednich punktach założyliśmy, że populacja zmierzonych okresów drgań wahadła pochodzi z rozkładu normalnego. Hipoteza ta wymaga jednak weryfikacji. Testem umożliwiającym weryfikację hipotezy, że dana populacja należy do danego rozkładu prawdopodobieństwa jest test χ^2. Statystyką testową jest:

$$\chi^2 = \sum_{i=1}^{k} \frac{(n_i - np_i)^2}{np_i}, \tag{15}$$

gdzie n_i – liczba elementów w i −tym przedziale, n − całkowita liczba elementów populacji, p_i − prawdopodobieństwo teoretyczne wylosowanie liczby w i −tym przedziale, k – liczba przedziałów. Jeśli wartość $\chi^2 \in [u(1-\alpha, k-1-\nu), \infty]$, to rozkład teoretyczny jest inny niż zakładany na poziomie istotności α. Zmienna $u(1-\alpha, k-1-\nu)$ to kwantyl rozkładu χ^2 o $k-1-\nu$ stopniach swobody, dla którego prawdopodobieństwo obliczone w przedziale $[-\infty, u(1-\alpha, k-1-\nu)]$ jest równe $1-\alpha$. Zmienna ν określa liczbę parametrów rozkładu, dla rozkładu normalnego $\nu = 2$, uwzględniając wartość oczekiwaną oraz odchylenie standardowe.

Dzięki narysowaniu histogramu uzyskujemy liczbę przedziałów k, liczbę elementów w danym przedziale n_i oraz granice poszczególnych przedziałów $h_{bin}^i = [h_l^i, h_r^i]$, gdzie h_l^i oraz h_r^i to pozycja odpowiednio lewej i prawej krawędzi $i-$ tego przedziału. Prawdopodobieństwo p_i można policzyć korzystając ze własności dystrubuanty (wzór (3)) jako

$$p_i = \Phi\left(\frac{(h_r^i - \mu)}{\sigma}\right) - \Phi\left(\frac{(h_l^i - \mu)}{\sigma}\right). \tag{16}$$

W programie Matlab wykorzystamy funkcję `erf` do obliczenia dystrubuanty rozkładu normalnego (wzór 4) oraz funkcję `chi2cdf` do obliczenia prawdopodobieństwa, że dana zmienna jest większa niż wartość statystyki χ^2. Jeśli prawdopodobieństwo jest większe niż poziom istotności α, to teoretyczny rozkład zgadza się z rozkładem populacji na poziomie istotności α. Kod programu przeprowadzający test χ^2 przy założeniu, że rozkład testowy to rozkład normalny prezentuje Listing 3.

```
clear
clc

dane=load('dane.txt');
n=length(dane);
mi_est=mean(dane);
std_est=std(dane);

h=2*iqr(dane)/n^(1/3);
T_max=max(dane);
T_min=min(dane);
k=round((T_max-T_min)/h);
[n_bin, h_bin]=hist(dane, k);
h_r=h_bin+(h_bin(2)-h_bin(1))/2;
h_l=h_bin-(h_bin(2)-h_bin(1))/2;
p_teor=(1+erf((h_r-mi_est)/sqrt(2)/std_est))/2-(1+erf((h_l-mi_est)/sqrt(2)/std_est))/2;
n_teor=n*p_teor;
Chi2=sum((n_bin-n_teor).^2./n_teor);
p_score=1-chi2cdf(Chi2, k-1-2);
```

Listing 3. Kod programu przeprowadzający test χ^2 dla zbioru testowego zawartego w pliku *dane.txt*, przy założeniu, że teoretyczny rozkład to rozkład normalny.

3.6. Test Kołmogorowa-Smirnowa

Test pozwalający na sprawdzenie czy dana populacja pochodzi z dowolnego ciągłego rozkładu prawdopodobieństwa jest test Kołmogorowa-Smirnowa. W teście tym porównuje się dystrybuantę teoretyczną oraz empiryczną, obliczaną na podstawie zebranej próbki zmiennej losowej. Załóżmy, że populacja X składa się z n próbek x z rozkładu ciągłego. Dystrybuantę empiryczną $F_n(x)$ oblicza się ze wzoru

$$F_n(x) = \frac{1}{n}\sum_{i-1}^{n} I_{x_i \leq x}, \tag{17}$$

Gdzie $I_{x_i \leq x}$ to funkcja równa 1, jeśli wartość zmiennej dla i −tej próbki jest mniejsza niż x. Algorytm może przebiegać następująco:

a) Należy posortować wektor próbek losowych x_i rosnąco

b) Znajdź unikalne wartości zmiennej losowej i przypisz je do nowej zmiennej wektorowej. Każdy element wektora to pojedyncza zmienna losowa

c) Dla każdego elementu wektora elementów unikalnych sprawdź ile jest zmiennych losowych mniejszych bądź równych niemu w pierwotnym wektorze zmiennych losowych i zsumuj je.

d) Przypisz do zmiennej wektorowej przechowującej wartości dystrybuanty empirycznej wartość sumy elementów mniejszych podzielonej przez całkowitą liczbę elementów próbki.

Implementację algorytmu w Matlabie prezentuje Listing 4.

```
[Y, I]=sort(dane, 'ascend');
dane=dane(I, :);
dane_uniq=unique(dane);
Dist_emp=zeros(length(dane_uniq), 1);
for i=1:length(dane_uniq)
    for j=1:n
        if dane(j)<=dane_uniq(i)
            Dist_emp(i)=Dist_emp(i)+1/n;
        end
    end
end
```

Listing 4. Implementacja algorytmu wyznaczania dystrybuanty empirycznej na bazie próby losowej.

Dystrybuanta empiryczna musi być porównana z dystrybuanta teoretyczną $F(x)$. Dystrybuantę można wyznaczyć dla każdego elementu wektora wartości unikalnych dzięki funkcji `normcdf`, jeśli teoretycznym rozkładem jest standaryzowany rozkład Gaussa. Jeśli próba losowa jest z rozkładu normalnego o dowolnej wartości oczekiwanej oraz odchyleniu standardowym, to funkcję `normcdf` należy wywołać dla zmiennej $\tilde{x} = \frac{x-\mu}{\sigma}$, gdzie rozkład zmiennej $\tilde{x}$ jest już standaryzowanym rozkładem normalnym.

W teście Kołmmogorowa-Smirnowa oblicza się statystykę

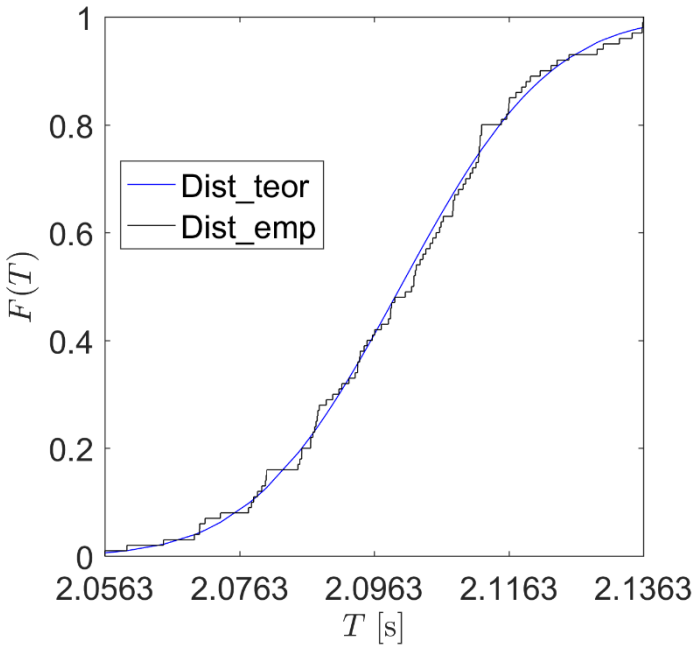
$$D_n = \sup_x |F_n(x) - F(x)|. \tag{18}$$

gdzie $\sqrt{n}D_n$ jest zmienną pochodzącą z rozkładu Kołmogorowa. Jeśli spełniony jest warunek $\sqrt{n}D_n > K_\alpha$, gdzie K_α jest kwantylem przedziału, dla którego obliczone prawdopodobieństwo jest równe $1-\alpha$. Program Matlab nie ma wbudowanej funkcji obliczającej prawdopodobieństwo z rozkładu Kołmogorowa, dlatego można użyć analitycznej postaci tej funkcji:

$$P(x \leq K) = \frac{\sqrt{2\pi}}{K} \sum_{i=1}^{\infty} \exp\left(\frac{(2i-1)^2\pi^2}{8K^2}\right), \tag{19}$$

gdzie pod K można wstawić obliczoną wartość $\sqrt{n}D_n$. Jeśli wartość $1 - P\left(x \leq \sqrt{n}D_n\right) < \alpha$, to należy odrzucić hipotezę, że rozkład populacji pochodzi z założonego rozkładu prawdopodobieństwa.

W programie Matlab jest wbudowana funkcja, która po przekazaniu do niej wektora zmiennej losowej zwraca wynik testu Kołmogorowa-Smirnowa. Funkcja ta nazywa się `kstest` i domyślnie przeprowadza test na poziomie istotności równym $\alpha = 0.05$. Wektor przekazywany do funkcji `kstest` musi być sprowadzony do standaryzowanego rozkładu normalnego. Na Rysunku 3. prezentowana jest dystrubuanta teoretyczna oraz empiryczna dla danych testowych używanych również w punktach 3.2-3.5. Kod programu Wykonujący Rysunek 3. Jak również przeprowadzający test Kołmogorowa-Smirnowa zarówno krok po kroku jak i z wykorzystaniem funkcji `kstest` prezentuje Listing 5.



Rysunek 3. Wykres dystrybuanty teoretycznej (*Dist_teor*) oraz dystrubyanty empirycznej (*Dist_emp*)

```
clear
clc

dane=load('dane.txt');
n=length(dane);
T_max=max(dane);
T_min=min(dane);
mi_est=mean(dane);
std_est=std(dane);

[Y, I]=sort(dane, 'ascend');
dane=dane(I, :);
dane_uniq=unique(dane);
Dist_emp=zeros(length(dane_uniq), 1);
for i=1:length(dane_uniq)
    for j=1:n
        if dane(j)<=dane_uniq(i)
            Dist_emp(i)=Dist_emp(i)+1/n;
        end
    end
end

Dist_teor=normcdf((dane_uniq-mi_est)/std_est);
D=max(abs(Dist_teor-Dist_emp));
Kol_var=sqrt(n)*D;

Kol_cdf=0;
x=Kol_var;
for i=1:300
    Kol_cdf=Kol_cdf+sqrt(2*pi)/x*exp(-(2*i-1)^2*pi^2/(8*x^2));
end
p_score_anal=1-Kol_cdf;

[decision, p_score]=kstest((dane-mi_est)/std_est);

figure
hold on
plot(dane_uniq, Dist_teor, '-b');
for i=1:length(dane_uniq)
    if i==1
    plot([dane_uniq(i) dane_uniq(i)], [0 Dist_emp(i)], '-k');
    else
    plot([dane_uniq(i-1) dane_uniq(i)], [Dist_emp(i-1) Dist_emp(i-1)], '-k');
    plot([dane_uniq(i) dane_uniq(i)], [Dist_emp(i-1) Dist_emp(i)], '-k');
    end
end
set(gca, 'Box', 'on', 'Fontsize', 15, 'Xtick', [T_min, ...
    (T_max-T_min)/4+T_min,(T_max-T_min)/2+T_min 3*(T_max-T_min)/4+T_min, T_max]);
xlabel('$T\; [\mathrm{s}$]', 'Interpreter', 'Latex', 'FontSize', 15);
ylabel('$F(T)$', 'Interpreter', 'Latex', 'FontSize', 15);
ylim([0, 1]);
xlim([T_min T_max]);
l=legend({'Dist\_teor', 'Dist\_emp'});
set(l, 'FontSize', 15, 'Position', [0.3 0.6 0.1 0.1]);
axis square
```

Listing 5. Kod programu wykonującego test Kołmogorowa-Smirnowa.

4. Podsumowanie

W ćwiczeniu przybliżono podstawowe rozkłady prawdopodobieństwa zmiennej ciągłej. Dla badanej populacji zmiennej losowej można wykonać szereg testów statystycznych pozwalających na sprawdzenie czy próbka pochodzi z hipotetycznego rozkładu prawdopodobieństwa. W szczegółach opisano test Studenta dla wartości oczekiwanej oraz dwa testy pozwalające ocenić czy rozkład populacji zgadza się z rozkładem modelowym. Rozważano wyłącznie próbę pochodzącą z rozkładu normalnego. W ćwiczeniu dodatkowo przybliżono metodę rysowania histogramu oraz dopasowania dowolnej krzywej teoretycznej do punktów pomiarowych. Obliczenia wykonano w programie Matlab.