

Dokumentacja programu w języku Python liczącego wydajność starterów wykorzystywanych do reakcji RT-qPCR – Primer Efficiency

1. Wprowadzenie

Program „Primer Efficiency” służy do liczenia wydajności starterów wykorzystywanych w metodzie RT-qPCR. Metoda RT-qPCR jest bardzo czułą metodą określania relatywnej zmiany poziomu wybranych transkryptów (<https://www.thermofisher.com/pl/en/home/brands/thermo-scientific/molecular-biology/molecular-biology-learning-center/molecular-biology-resource-library/spotlight-articles/basic-principles-rt-qpcr.html>).

Do uzyskania wiarygodnych rezultatów kluczowym jest etap projektowania i optymalizacji starterów (ang. *primers*). Startery powinny być specyficzne dla badanego transkryptu, namnażany fragment powinien mieć optymalną długość. Startery należy również przetestować pod kontem ich wydajności, ponieważ w dalszych obliczeniach określania relatywnej zmiany poziomu transkryptów zakładamy, że startery dla porównywanych genów mają jednakową wydajność. Rekomenduje się wykorzystywanie starterów, których wydajność leży w przedziale 90-100%. Wydajność określa się poprzez wyznaczenie krzywej standardowej uzyskanej wartości Ct względem rozcieńczenia matrycy. Następnie oblicza się wydajność starterów używając współczynnika nachylenia osi (a) korzystając z równania : $10^{(-1/a)-1}$. Skrypt analizuje dane z arkuszy kalkulacyjnych Excel, wyznacza regresje liniową, oblicza wydajności starterów i przedstawienie wyniki na wykresie oraz w tabeli.

2. Szczegółowy opis wykorzystanych funkcji i działania programu

1) Import bibliotek

Program napisany jest z użyciem bibliotek Pandas, NumPy i Matplotlib

```
1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
```

2) Określenie ścieżki do pliku Excel i otwarcie pliku

Użytkownik z poziomu terminala proszony jest o podanie ścieżki do pliku z danymi do analizy w programie Excel.

Plik Excel zostaje otwarty z wykorzystaniem biblioteki Pandas.

```
9 file_path = input(r"Enter the path to the Excel file: ")
10
11 # Otwarcie pliku w Excelu
12 xls = pd.ExcelFile(file_path)
```

Aby program odpowiednio działał plik powinien zostać sformatowany w następujący sposób:

- Dane dla każdego startera znajdują się w osobnym arkuszu skoroszytu
- Nazwa startera = nazwa arkusza
- w kolumnie o tytule: "Template Dilution" → rozcieńczenie matrycy
- w kolumnie o tytule: "Ct Value" → wartość ct dla określonego rozcieńczenia

Przykład:

C17						
	A	B	C	D	E	F
1		Template Dilution	Ct Value			
2	1	0,5	27,5941			
3	2	0,25	28,3189			
4	3	0,125	29,826			
5	4	0,0625	30,8687			
6	5	0,03125	31,4804			
7	6	0,015625	32,0509			

Primer_rps Primer_ef1a Primer_M2v3 **Primer_M2**

- 3) Pobrane zostają nazwy arkuszy kalkulacyjnych, czyli nazwy starterów. Utworzona zostaje DataFrame, w którym zostaną zapisane obliczone wartości wydajności starterów. Stworzona zostaje figura z wykresem w której zostaną zaprezentowane dane. Wykorzystana została biblioteka Matplotlib.

```
14 # Pobranie nazw starterów z arkusza Excel
15 primer_names = xls.sheet_names
16
17 # Utworzenie Pandas DataFrame do przechowywania danych o efektywności starterów
18 efficiency_df = pd.DataFrame(columns = ["Primer Name", "Efficiency"])
19
20 # Utworzenie figury i osi na wykres z Matplotlib
21 fig, ax = plt.subplots() #funkcja tworzy figure oraz osie
```

- 4) Kod wykonuje analizę danych dla każdego startera w pętli:

```
25 for primer_name in primer_names:
    • Dane z arkusza są załadowywane z wykorzystaniem modułu Pandas
    • Obliczane są logarytmy dziesiętne rozcieńczenia matrycy (kolumna „Template Dilution”), które są wartościami x. Wartości x umieszczane są w macierzy x_values
    • Wartości y są pobierane z kolumny „Ct Value”. Umieszczone zostają w macierzy y_values
        27 Excel_df = pd.read_excel(xls, sheet_name = primer_name) #
        28
        29 x_values = np.log10(Excel_df['Template Dilution'].values)
        30 y_values = Excel_df['Ct Value'].values #tworzy macierz

    • Punkty określone przez współrzędne x i y zostają umieszczone na wykresie.
        ax.scatter(x_values, y_values, label= f'{primer_name}')

    • Zdefiniowana zostaje funkcja regresji liniowej ( $y = ax + b$ ). Dane zostają dopasowane do funkcji z wykorzystaniem modułu Numpy. W wyniku pozyskane zostają parametry a i b dla funkcji.

36 def regression_function(x, a, b):
37     return a * x + b

40 a_parameter, b_parameter = np.polyfit(x_values, y_values, 1)
```

- Wyznaczona zostaje miara jakości dopasowania modelu funkcji liniowej do danych $-R^2$.

```

43 y_predicted = regression_function(x_values, a_parameter, b_parameter)
44 rss = ((y_values - y_predicted)**2).sum() #RSS (sum of squares of r
45 tss = ((y_values - y_values.mean())**2).sum() #TSS (total sum of sq
46 r2 = 1 - (rss/tss)

```

- Obliczona zostaje wydajność starterów z równania: $10^{(-1/a)} - 1$
Dodanie wyliczonych wartości do DataFrame

```

48 # Obliczenie wydajności starterów równaniem: 10^(-1/a) - 1
49 efficiency = "{:.2%}".format(10 ** (-1 / a_parameter) - 1)
50 # dodanie wyliczonych wartości wydajności do dataframe
51 efficiency_df = pd.concat([efficiency_df, pd.DataFrame({"Primer Name": [primer_name], "Efficiency": [efficiency]})])

```

- Do wykresu dodane zostaje krzywa dopasowania liniowego

```

54 plt.plot(x_values, y_predicted, linestyle = "dashed")

```

- Informacje o regresji, R^2 i wydajności dla poszczególnych starterów są dodawane do figury

```

56 fig.text(0.85, n, f'{primer_name}'
57          f'\ny = {a_parameter:.4f}x + {b_parameter:.4f}'
58          f'\nR^2 = {r2:.4f} \nEfficiency={efficiency}\n\n')

```

5) Dopasowanie, opisanie wykresu, osi, etc. Pokazanie wykresu

```

64 fig.set_size_inches(12,10)
65 ax.set_xlabel('log10(Template Concentration)')
66 ax.set_ylabel('Ct Value')
67 ax.set_title('Ct values for specific primers in template dilu
68 ax.legend()
69 ax.grid(True, color = '#a4a9ad')
70 plt.subplots_adjust(right=0.8)
71 ax.set_ylim(0)
74 plt.show()

```

6) Wydrukowanie tabeli z wydajnością starterów w terminalu

```

77 print("\nPrimer Efficiency:")
78 print(efficiency_df)

```

3. Przykład działania programu:

Po uruchomieniu programu użytkownik proszony jest o podanie ścieżki do pliku.

Enter the path to the Excel file:

Przykładowa ścieżka: C:\Users\joann\OneDrive\Pulpit\primer_efficiency_raw_data.xlsx

W oknie Terminala zostaje wydrukowana tabela z wydajnościami dla starterów:

```

Primer Efficiency:
Primer Name Efficiency
0 Primer_rps      89.60%
0 Primer_ef1a     91.59%
0 Primer_M2v3    209.00%
0 Primer_M2     109.47%

```

Zostaje utworzony wykres na którym znajduje się graficzne przedstawienie danych z arkusza, dopasowanie funkcji liniowej, wartości dopasowania funkcji oraz wartość wydajności starterów.

