

An R package to consolidate pretest probability models and guidelines for CAD

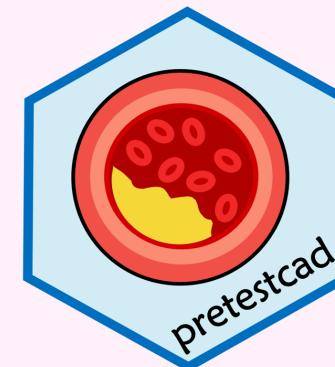
Jeremy Selva 

@JauntyJJS    

<https://jeremy-selva.netlify.app> 

For R/Pharma 2025 

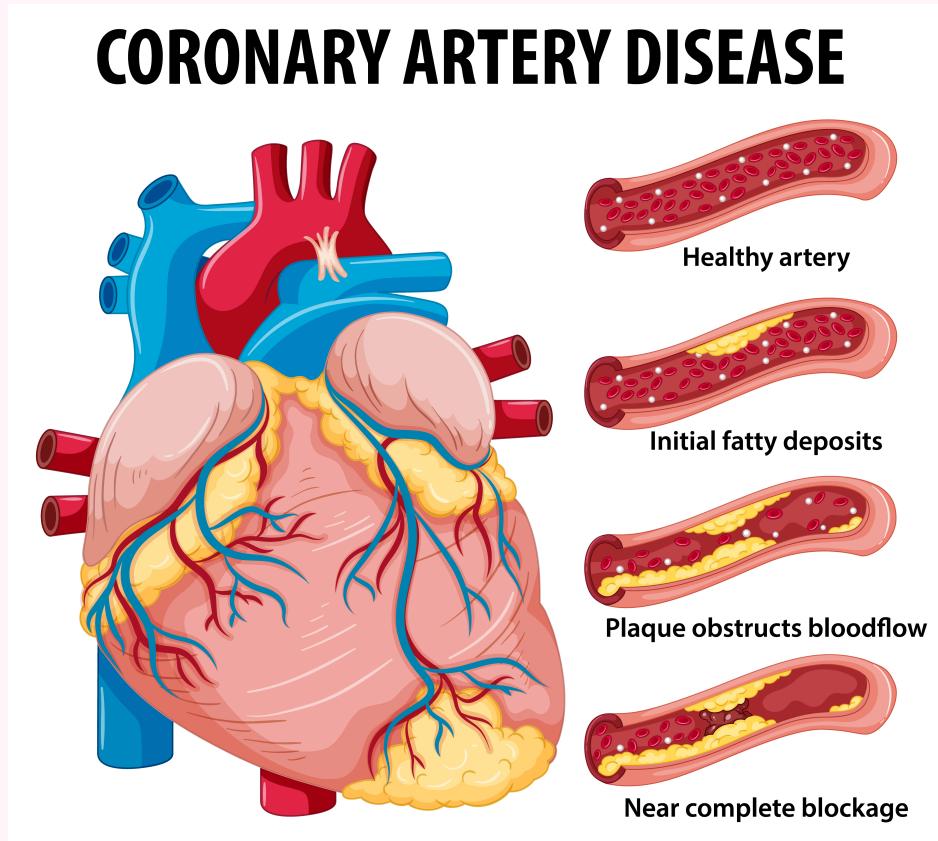
4th November 2025



What is Coronary Artery Disease (CAD) ?

Coronary Artery Disease (CAD) happens when the coronary arteries from the heart become narrow or blocked by fatty deposits called plaque.

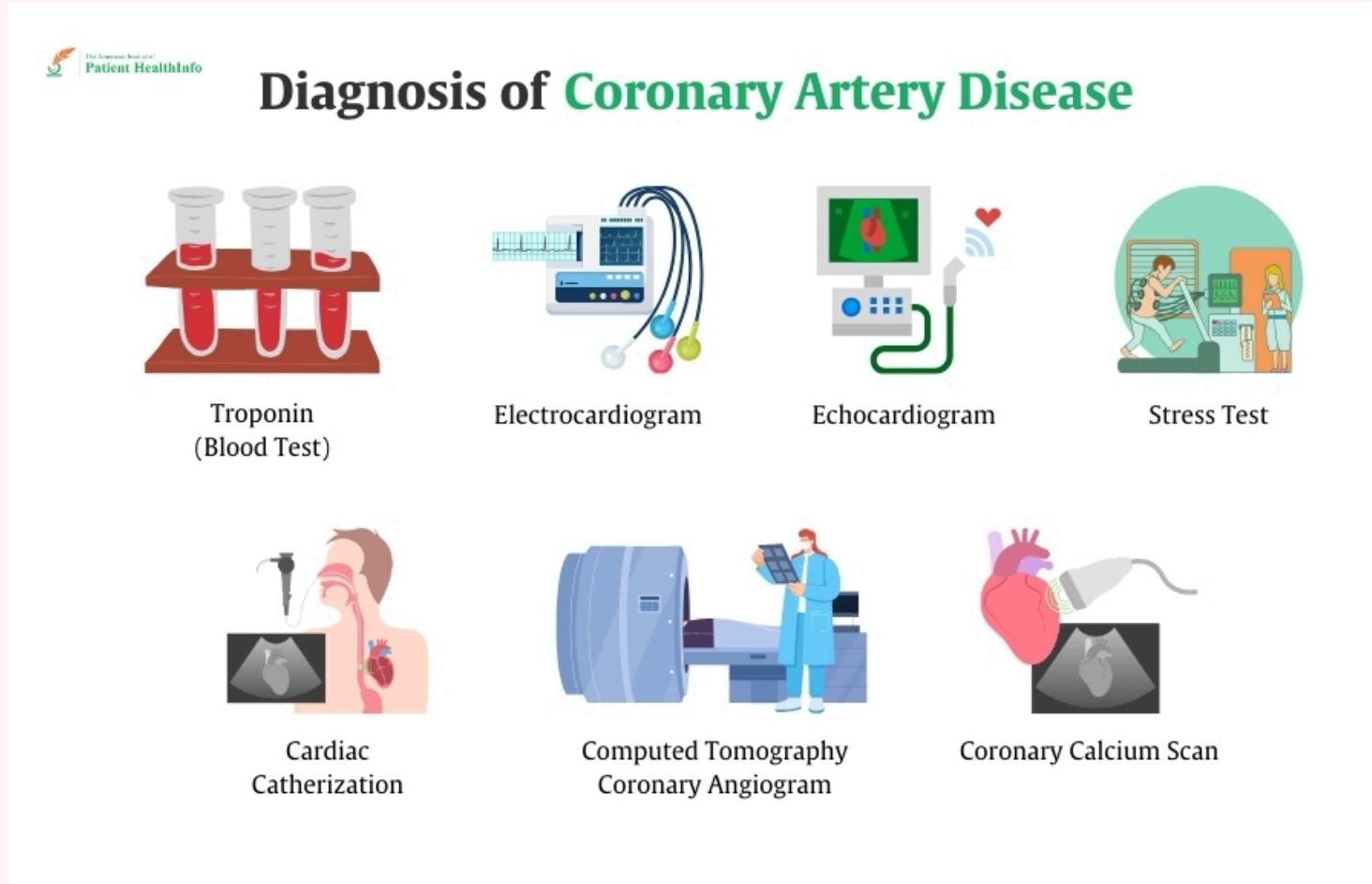
This can lead to chest pain, shortness of breath or even heart attack.



[Coronary Artery Disease for health education Infographic](#) designed by [brgfx](#) from Freepik.

Diagnosis of CAD

Patient needs to go for many invasive, time consuming and expensive tests for diagnosis of CAD.

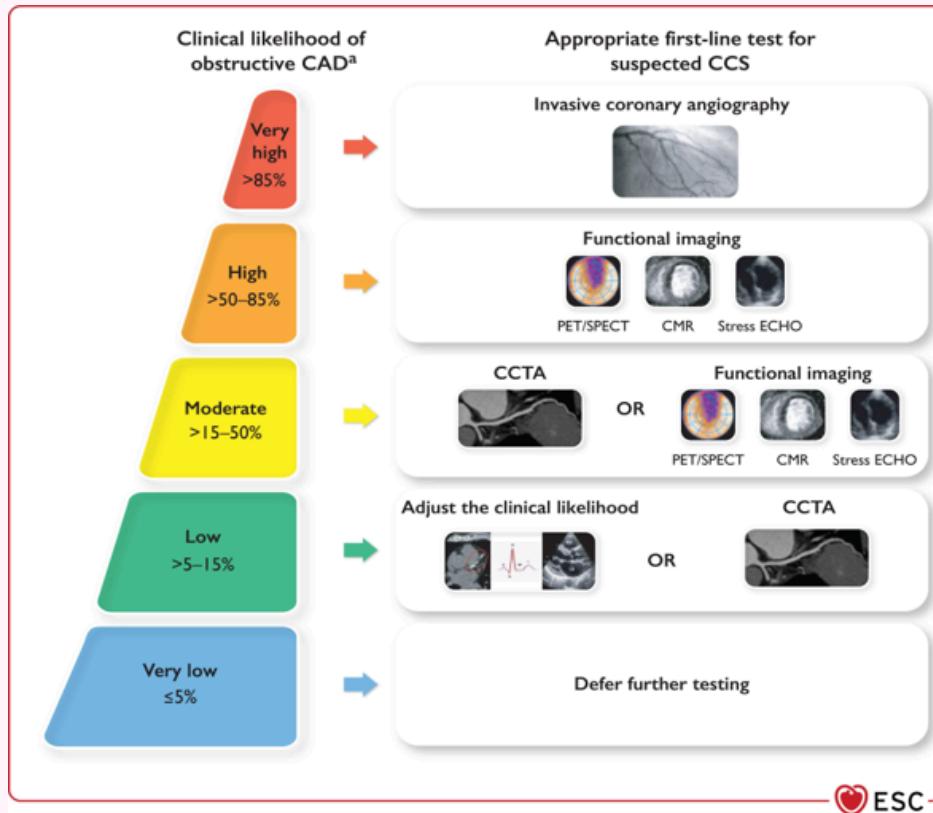


Images from Pokharel, MD, B., & Dhakal, MD, B. *The American Journal of Patient Health Info* 2024; 1(01) doi: [10.69512/ajphi.v1i01.37](https://doi.org/10.69512/ajphi.v1i01.37).

What is Pretest Probability ?

Pretest probability is the estimated probability, given a set of risk factors (e.g age, sex, etc.), that a patient has a specific disease or condition before any diagnostic tests are performed.

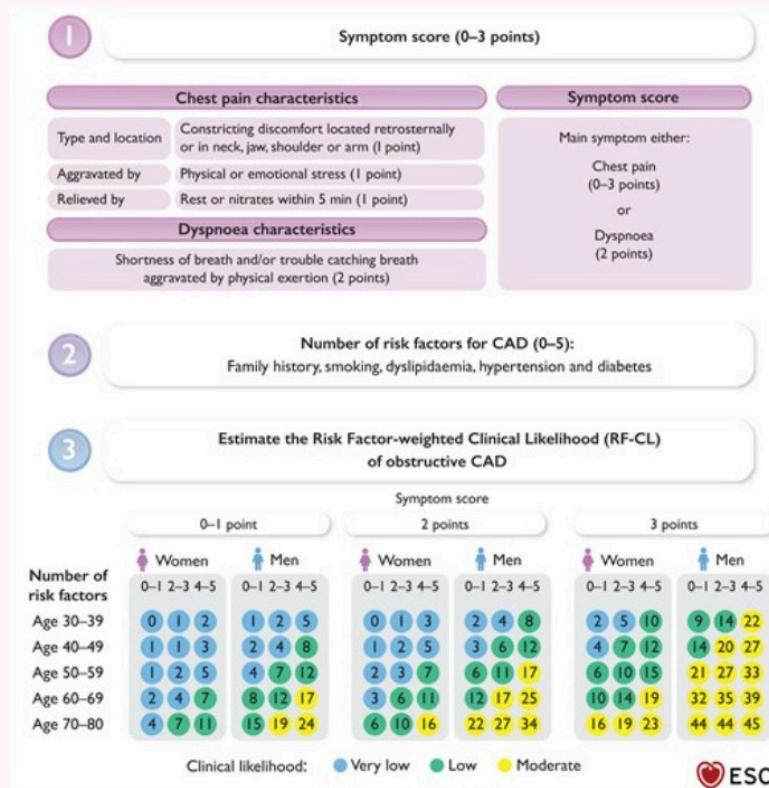
It helps clinicians and doctors makes informed decision on which patients go for which diagnostic tests.



[European Society of Cardiology \(ESC\) 2024 guidelines](#) of first-line testing in symptomatic individuals with suspected chronic coronary syndrome.

Pretest Probability Calculation

Pretest probability of CAD may be presented in the form of a table or an online calculator where we calculate the probability for each patient.



The screenshot shows the QxMD Calculate app interface with the following details:

- Calculator by QxMD**
- Navigation:** All Calculators, Calculator (selected), About, References.
- Title:** Pre-test probability of CAD (CAD consortium)
- Description:** Determine pre-test probability of coronary artery disease in patients with chest pain.

Workflow from Vrints et al. *European Heart Journal* 2024; 45(36):3415-3537 doi:

[10.1093/eurheartj/ehae177](https://doi.org/10.1093/eurheartj/ehae177).

Motivation

As pretest models changes with time, it becomes tedious to calculate and update pretest probability one patient at a time.



[Burntout girl](#) by [Freepik](#).

[Crowd of people wearing face masks](#) designed by [Freepik](#).

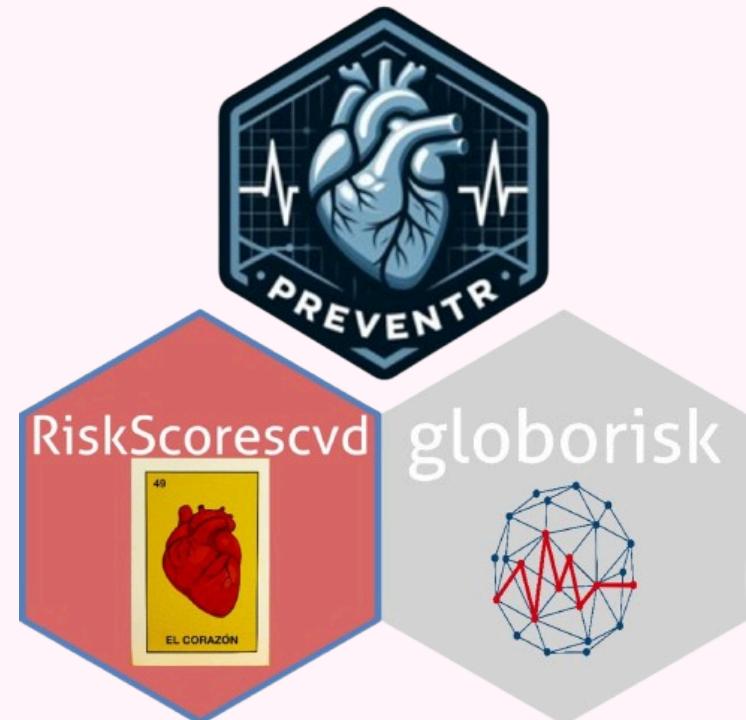
Motivation

There are many R packages for calculating risk for cardiovascular disease (CVD).

Why not create one for CAD ?



[Motivated guy ready to start work](#) by [Freepik](#).



R packages [preventr](#), [RiskScorecvd](#) and [globorisk](#) to calculate risk of CVD.

pretestcad

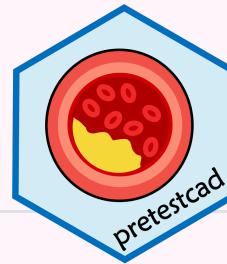
<https://jauntyjjs.github.io/pretestcad/>

pretestcad 1.1.0 Reference Changelog

Search for



pretestcad



R package used to calculate different **PreTest Probability** (PTP) scores for obstructive **Coronary Artery Disease** (CAD).

As diagnosis of CAD involves a costly and invasive coronary angiography procedure for patients, having a reliable PTP for CAD helps doctors to make better decisions during patient management. This ensures high risk patients can be diagnosed and treated early for CAD while avoiding unnecessary testing for low-risk patients.

Table of Content

- [Installation](#)
- [Currently available pretest probability scores](#)

Links

[View on CRAN](#)

[Browse source code](#)

[Report a bug](#)

License

[Full license](#)

[MIT + file LICENSE](#)

Citation

[Citing pretestcad](#)

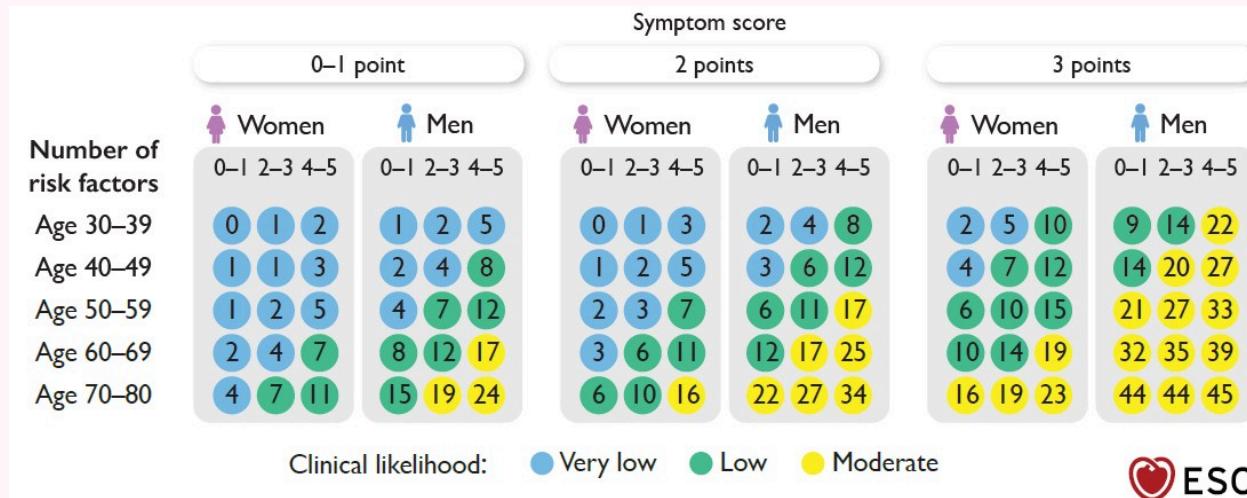
Developers

Jeremy Salva

Usage Example

```
1 # 30 female with symptom score of 3 and 0 risk factors
2 calculate_esc_2024_fig_4_ptp(
3   age = 30,
4   sex = "female",
5   chest_pain_type = "typical",
6   have_dyspnoea = "no",
7   have_family_history = "no",
8   have_smoking_history = "no",
9   have_dyslipidemia = "no",
10  have_hypertension = "no",
11  have_diabetes = "no",
12  output = "percentage"
13 )
```

```
[1] "2%"
```

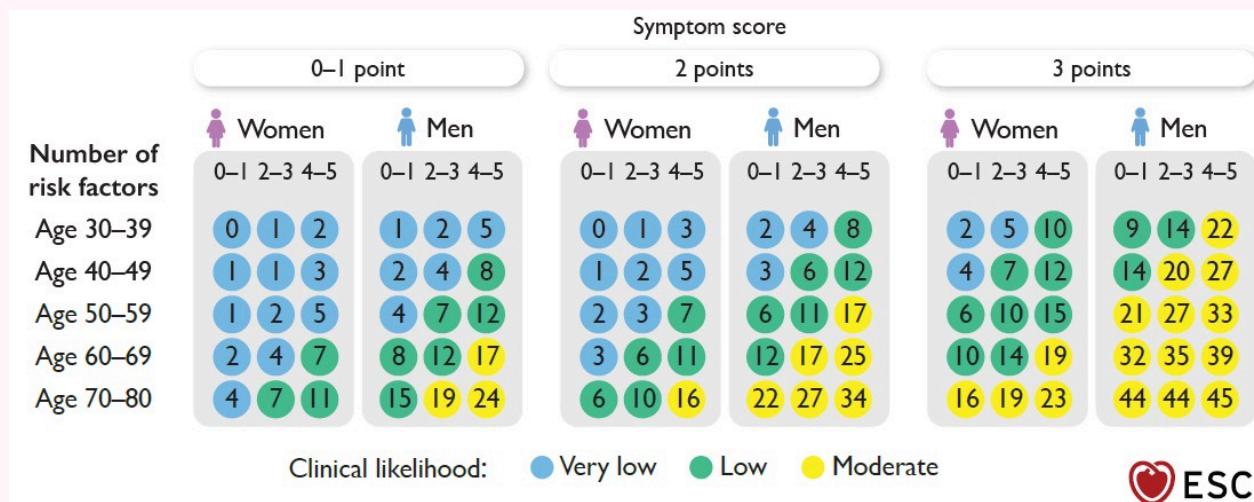


ESC 2024 guidelines from Vrints et al. *European Heart Journal* 2024; 45(36):3415-3537 doi: [10.1093/eurheartj/ehae177](https://doi.org/10.1093/eurheartj/ehae177).

Robust to missing clinical data

```
1 # 55 male with symptom score of 3 and 3 risk factors and 2 NA
2 calculate_esc_2024_fig_4_ptp(
3   age = 55,
4   sex = "male",
5   chest_pain_type = "typical",
6   have_dyspnoea = "no",
7   have_family_history = NA,
8   have_smoking_history = NA,
9   have_dyslipidemia = "yes",
10  have_hypertension = "yes",
11  have_diabetes = "yes",
12  max_na_num_of_rf = 2,
13  output = "percentage"
14 )
```

```
[1] "27%"
```



ESC 2024 guidelines from Vrints et al. *European Heart journal* 2024; 45(36):3415-3537 doi: [10.1093/eurheartj/ehae177](https://doi.org/10.1093/eurheartj/ehae177).

Helpful error messages

Applied this [talk](#) from useR! 2024 to write better error messages.

Making Better Error Messages with Rlang and Cli - Emil Hvitfeldt



Helpful error messages

```
1 calculate_esc_2024_fig_4_ptp(  
2   age = "Something Else",  
3   sex = "male",  
4   chest_pain_type = "typical",  
5   have_dyspnoea = "no",  
6   have_family_history = "no",  
7   have_smoking_history = "no",  
8   have_dyslipidemia = "yes",  
9   have_hypertension = "yes",  
10  have_diabetes = "yes",  
11  output = "percentage"  
12 )
```

```
Error in `calculate_esc_2024_fig_4_ptp()`:  
! Provided input `age`, must be <numeric>, `NA` or `NaN`. It is  
currently "Something Else" of type <character>
```

```
1 calculate_esc_2024_fig_4_ptp(  
2   age = 0,  
3   sex = "male",  
4   chest_pain_type = "typical",  
5   have_dyspnoea = "no",  
6   have_family_history = "no",  
7   have_smoking_history = "no",  
8   have_dyslipidemia = "yes",  
9   have_hypertension = "yes",  
10  have_diabetes = "yes",  
11  output = "percentage"  
12 )
```

```
Error in `calculate_esc_2024_fig_4_ptp()`:  
! `age` must be positive, not 0
```

```
1 calculate_esc_2024_fig_4_ptp(  
2   age = 50,  
3   sex = "Something Else",  
4   chest_pain_type = "typical",  
5   have_dyspnoea = "no",  
6   have_family_history = "no",  
7   have_smoking_history = "no",  
8   have_dyslipidemia = "yes",  
9   have_hypertension = "yes",  
10  have_diabetes = "yes",  
11  output = "percentage"  
12 )
```

```
Error in `calculate_esc_2024_fig_4_ptp()`:  
! `sex` must be one of "male", "female", "NA", or "NaN", not "Something  
Else".
```

```
1 calculate_esc_2024_fig_4_ptp(  
2   age = 50,  
3   sex = "ale",  
4   chest_pain_type = "typical",  
5   have_dyspnoea = "no",  
6   have_family_history = "no",  
7   have_smoking_history = "no",  
8   have_dyslipidemia = "yes",  
9   have_hypertension = "yes",  
10  have_diabetes = "yes",  
11  output = "percentage"  
12 )
```

```
Error in `calculate_esc_2024_fig_4_ptp()`:  
! `sex` must be one of "male", "female", "NA", or "NaN", not "ale".  
i Did you mean "male"?
```

Use of purrr

```
1 patient_data <- tibble::tribble(
2   ~unique_id,
3   ~age,      ~sex,
4   ~chest_pain_type, ~have_dyspnoea,
5   ~have_family_history, ~have_smoking_history, ~have_dyslipidemia, ~have_hypertension, ~have_diabetes,
6   "45 year old male with typical chest pain, no dyspnoea, hypertension and diabetes",
7   45, "male",
8   "typical", "no",
9   "no", "no", "no", "yes", "yes",
10  "70 year old female with no chest pain, dyspnoea, have smoking history (past or current smoker) and dyslipidemia",
11  70, "female",
12  "no chest pain", "yes",
13  "no", "yes", "yes", "no", "no"
14 )
15
16 risk_data <- patient_data |>
17   dplyr::mutate(
18     esc_2024_ptp_percent = purrr::pmap_chr(
19       .l = list(
20         age = .data[["age"]],
21         sex = .data[["sex"]],
22         chest_pain_type = .data[["chest_pain_type"]],
23         have_dyspnoea = .data[["have_dyspnoea"]],
24         have_family_history = .data[["have_family_history"]],
25         have_smoking_history = .data[["have_smoking_history"]],
26         have_dyslipidemia = .data[["have_dyslipidemia"]].
```

```
# A tibble: 2 × 2
#>   unique_id          esc_2024_ptp_percent
#>   <chr>                <chr>
#> 1 45 year old male with typical chest pain, no dyspnoea, h... 20%
#> 2 70 year old female with no chest pain, dyspnoea, have sm... 10%
```

Flexible Labeling

Applied this [talk](#) from R/Medicine 2025 to include flexible labelling.

kidney.epi R Package for Facilitating Research in Diabetes, Kidney, Heart, and Other Diseases



Flexible Labeling



kidney.epi egfr.{...} arguments: sex

- Custom labels for males and females used in a data set could be defined as arguments

```
# label(s) used to define female sex in the dataset  
label_sex_female = c("Female"),
```

- Custom labels reflect the definitions used in the data set
 - numeric label: if the data define sex as 0, 1
 - string label: “F”, “Female”, “f”
 - SNOMED CT code: 248152002
 - LOINC code: “LA3-6”
 - a code from any other ontology

Flexible Labeling

```
1 calculate_esc_2024_fig_4_ptp(  
2   age = 50,  
3   sex = "男性",  
4   chest_pain_type = "typical",  
5   have_dyspnoea = "tidak",  
6   have_family_history = "no",  
7   have_smoking_history = "no",  
8   have_dyslipidemia = "yes",  
9   have_hypertension = "yes",  
10  have_diabetes = "ஆம்",  
11  output = "percentage",  
12  label_sex_male = c("male", "男性", "男人"),  
13  label_have_dyspnoea_no = c("no", "tidak"),  
14  label_have_diabetes_yes = c("yes", "ஆம்", "ஆமாம்")  
15 )
```

```
[1] "27%"
```

```
1 calculate_esc_2024_fig_4_ptp(  
2   age = 50,  
3   sex = "女性性",  
4   chest_pain_type = "typical",  
5   have_dyspnoea = "no",  
6   have_family_history = "no",  
7   have_smoking_history = "no",  
8   have_dyslipidemia = "yes",  
9   have_hypertension = "yes",  
10  have_diabetes = "no",  
11  output = "percentage",  
12  label_sex_male = c("male", "男性", "男人"),  
13  label_sex_female = c("female", "女性", "女人"),  
14  label_sex_unknown = c("NIL", NA, NaN)  
15 )
```

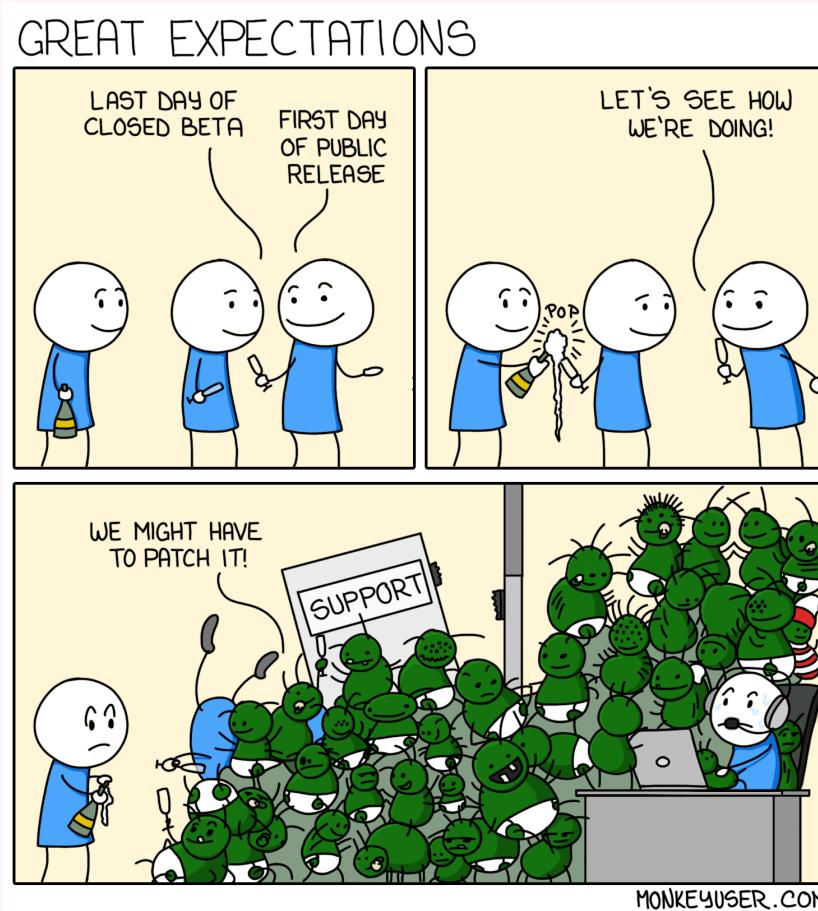
```
Error in `calculate_esc_2024_fig_4_ptp()`:  
! `sex` must be one of "male", "男性", "男人", "female", "女性", "女人",  
  "NIL", "NA", or "NaN", not "女性性".  
i Did you mean "女性"?
```

```
1 calculate_esc_2024_fig_4_ptp(  
2   age = 50,  
3   sex = "male",  
4   chest_pain_type = "typical",  
5   have_dyspnoea = "no",  
6   have_family_history = "no",  
7   have_smoking_history = "no",  
8   have_dyslipidemia = "yes",  
9   have_hypertension = "yes",  
10  have_diabetes = "no",  
11  output = "percentage",  
12  label_sex_male = c("Anything", "male", "男性", "男人", "随便"),  
13  label_sex_female = c("Anything", "male", "女性", "女人"),  
14  label_sex_unknown = c("Anything", "NIL", NA, NaN, "随便")  
15 )
```

```
Error in `calculate_esc_2024_fig_4_ptp()`:  
! `label_sex_male`, `label_sex_female` and `label_sex_unknown` must be  
  mutually exclusive.  
Common values found in `label_sex_male` and `label_sex_female`: "Anything" and  
  "male".  
Common values found in `label_sex_male` and `label_sex_unknown`: "Anything" and  
  "随便".  
Common values found in `label_sex_female` and `label_sex_unknown`: "Anything".  
Please ensure `label_sex_male`, `label_sex_female` and `label_sex_unknown` do  
  not hold common values.
```

Thank you

Available on [CRAN](#), [RUniverse](#) and [Github](#).



[Great Expectations](#) from [MonkeyUser.com](#)