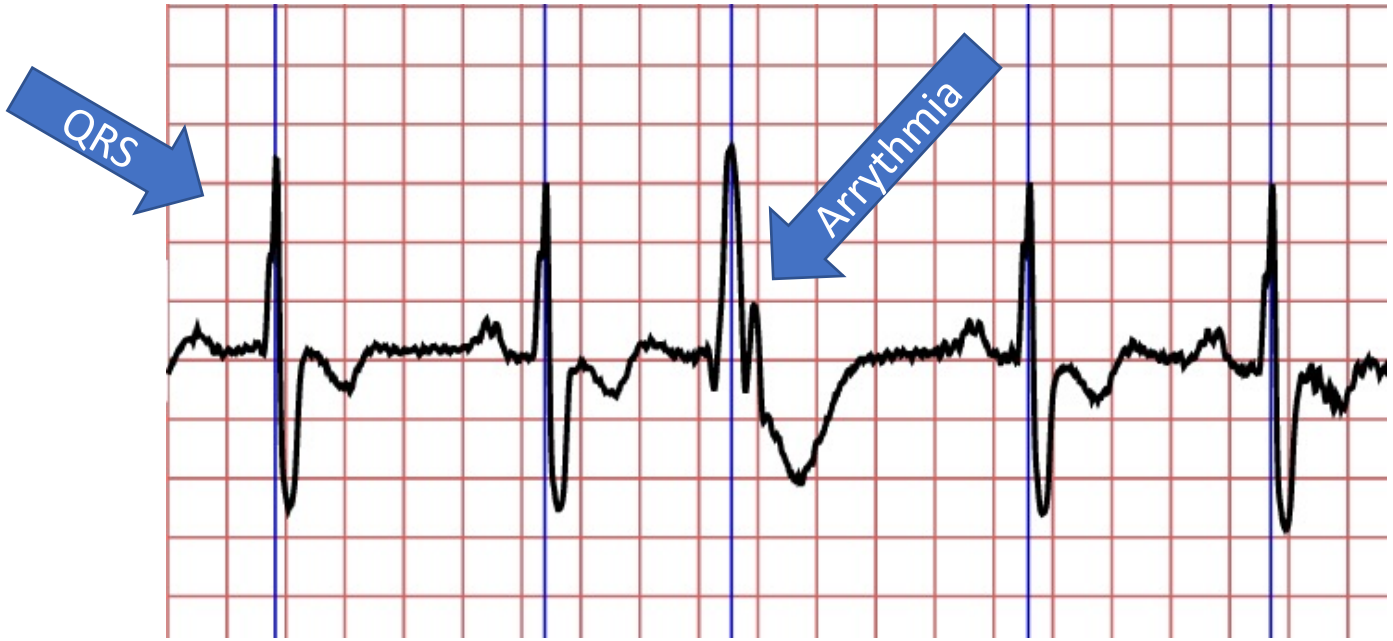
ENGR 298: Engineering Analysis and Decision Making – Evaluation Metrics

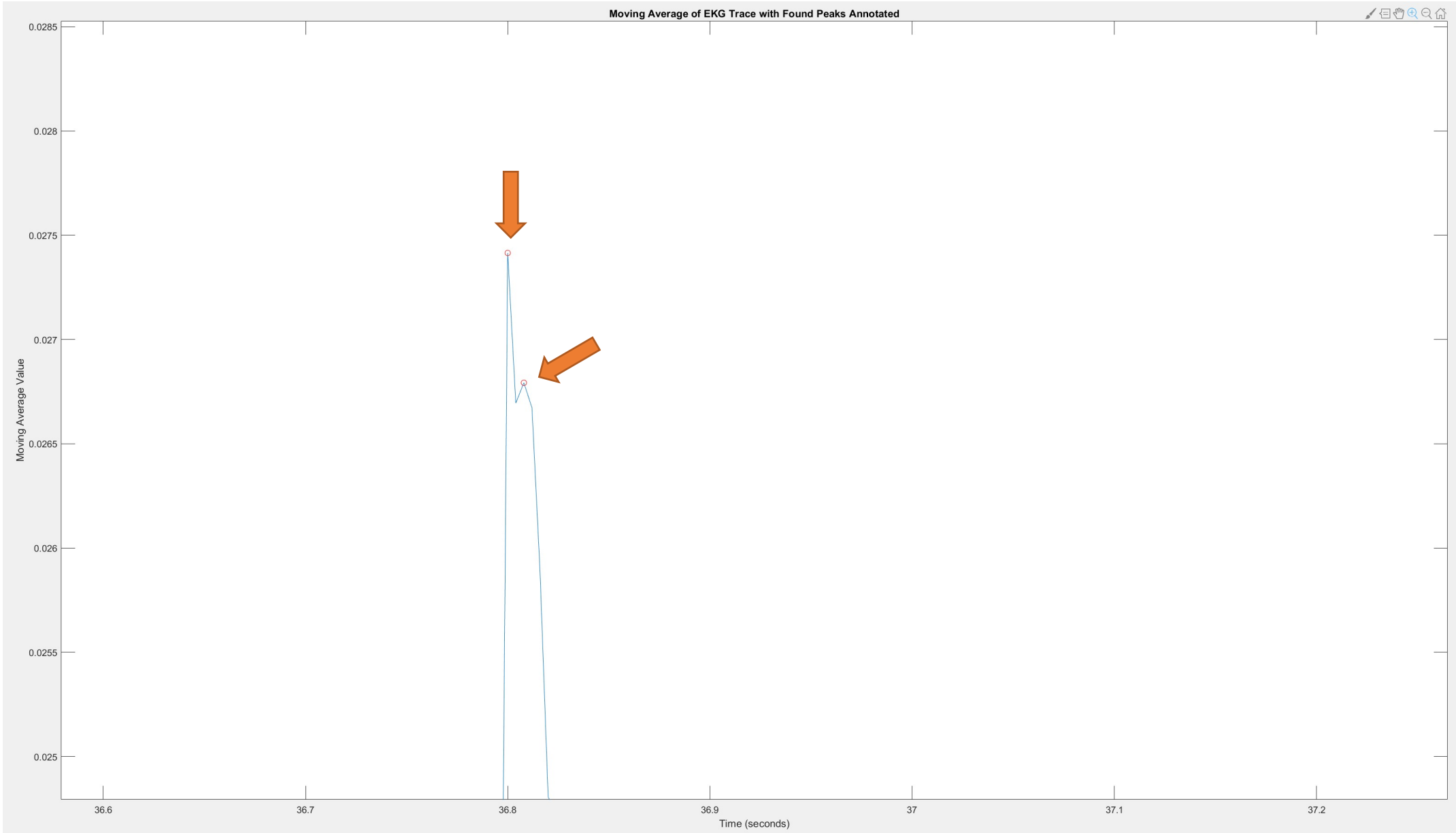
How to know you're right when you're probably not… ☺

Dr. Jason Forsyth
Department of Engineering
James Madison University

How accurate are you?

- You’re developing a detection algorithm for heart beats. The goal is to distinguish **positive/true** events in the signal from **negative/false** events.




Moving Average of EKG Trace with Found Peaks Annotated
0.0285
0.028
0.0275
0.027
0.0265
0.026
0.0255
0.025
Moving Average Value
36.6
36.7
36.8
36.9
37
37.1
37.2
Time (seconds)

Some Definitions

- **True Positive**: correctly identifying the item of interest is present
- **True Negative**: correctly identifying the item of interest is not present.
- **False Positive**: an incorrect detection; event is not actually present.
- **False Negative**: a missed detection; event occurred but was not identified.

Some Metrics

- $Sensitivity = True\ Positive\ Rate = \frac{TP}{TP+FN}$

  How many positives were found. High sensitivity means all were found (but could also have many FP)

- $Specificity = True\ Negative\ Rate = \frac{TN}{TN+FP}$
  - What percentage of the negative were found versus missed?
- $Precision = \frac{TP}{TP+FP}$
  - How much can we rely on the Positive results? How many positives were correctly identified?

- $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$
  - Across all tests (positive and negative), how many were correctly identified/

On Metrics

- A diagnostic system doesn’t need to be perfect.

- Consider if the treatment for some disease will cure you. But if you’re not sick then nothing will happen. The test should have a Sensitivity (we need all positive cases), event if we get some extra positives.

- The distribution of positives and negatives is important as well. If an event is rare (1 in a 1,000,000), and it’s missed, then you still got (999,999 / 1,000,000) correct for a high accuracy ☺

How accurate are test results?

No test gives a 100% accurate result; tests need to be evaluated to determine their sensitivity and specificity, ideally by comparison with a "gold standard." The lack of such a clear-cut "gold-standard" for covid-19 testing makes evaluation of test accuracy challenging.

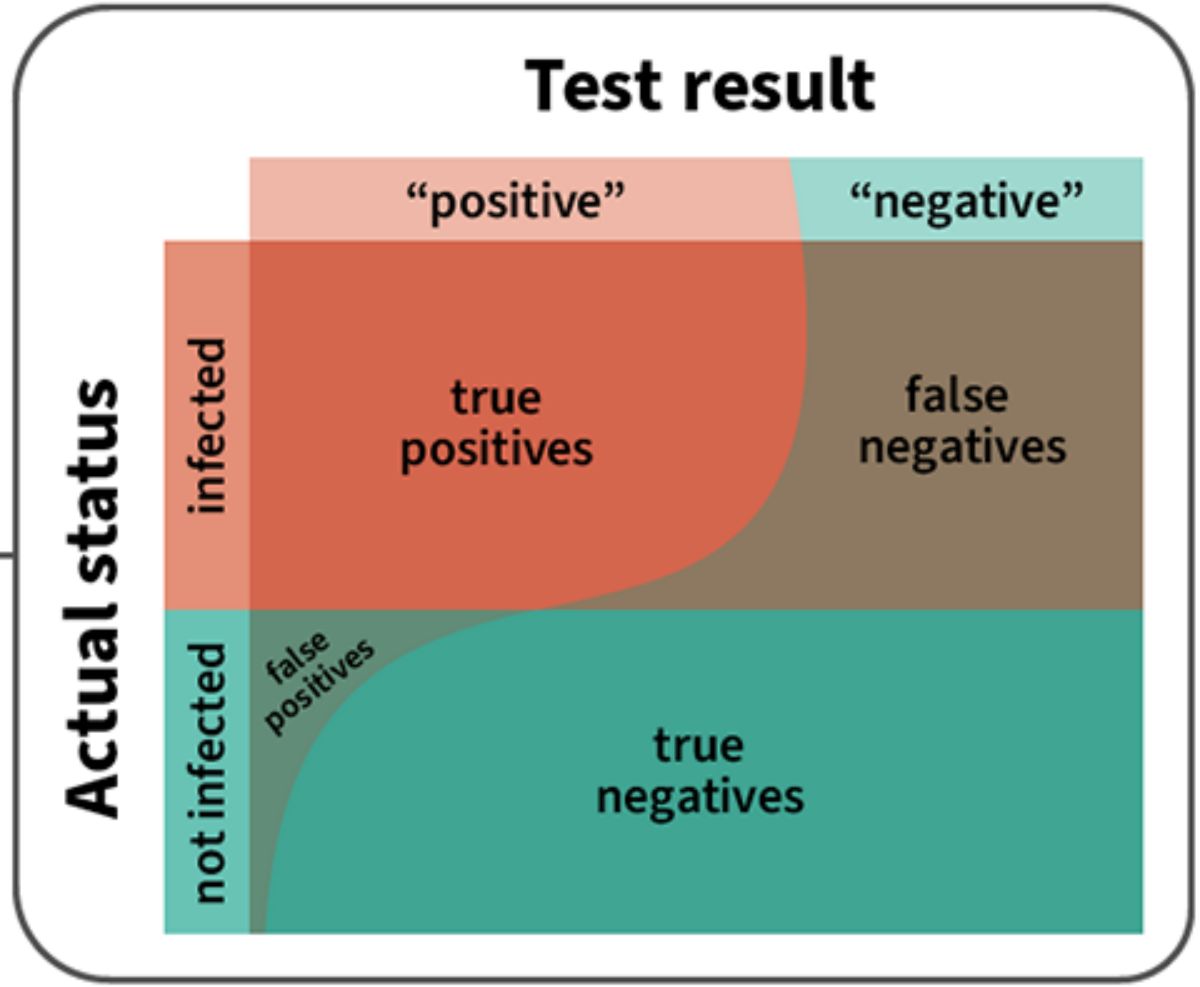
A systematic review of the accuracy of covid-19 tests reported false negative rates of between 2% and 29% (equating to sensitivity of 71-98%), based on negative RT-PCR tests which were positive on repeat testing.[6] The use of repeat RT-PCR testing as gold standard is likely to underestimate the true rate of false negatives, as not all patients in the included studies received repeat testing and those with clinically diagnosed covid-19 were not considered as actually having covid-19.[6]

Accuracy of viral RNA swabs in clinical practice varies depending on the site and quality of sampling. In one study, sensitivity of RT-PCR in 205 patients varied, at 93% for broncho-alveolar lavage, 72% for sputum, 63% for nasal swabs, and only 32% for throat swabs.[7] Accuracy is also likely to vary

https://www.bmj.com/content/bmj/369/bmj.m1808.full.pdf

The COVID-19 swab test is highly **specific** but not as **sensitive.**

That means a positive result is almost always true, but a negative result is sometimes false.



Sensitivity = number of **true positives** / number of those tested who **really are infected** = **"how many of the infections did we find?"**

Specificity = number of **true negatives** / number of those tested who **really are not infected** = **"how many of the healthy people did we clear?"**

https://medical.mit.edu/covid-19-updates/2020/06/how-accurate-diagnostic-tests-covid-19

But the accuracy of even these tests depends on the percentage of people in the population who have actually been exposed to the virus. For example, in a population where the prevalence of infection is 5%, a test with 95% specificity and 95% sensitivity will return the same number of false positives as true positives, making any individual result no more useful than the flip of a coin.

https://medical.mit.edu/faqs/faq-testing-covid-19#faq-12

COVID IgG testing 95% is sensitivity 90% specificity

What is the predictive value?

Prevalence 5%

	(+)	(−)	
Truth	50	950	1000
(+)	48	95	143
(−)	2	855	857

$PPV = \frac{48}{143} \Rightarrow 34\%$

$NPV = \frac{855}{857} \Rightarrow 99.8\%$

Prevalence 30%

	(+)	(−)	
Truth	300	700	
(+)	285	70	355
(−)	15	630	645

$PPV = \frac{285}{355} = 80\%$

$NPV = \frac{630}{645} = 97.7\%$

https://www.youtube.com/watch?v=x5FfTjJtV3E

Table 3. Relationship between pre-test probability and the likelihood of positive and negative predictive values

Pretest Probability*	Negative Predictive Value**	Positive Predictive Value**	Impact on Test Results
Low	High	Low	Increased likelihood of False Positives Increased likelihood of True Negatives
High	Low	High	Increased likelihood of True Positives Increased likelihood of False Negatives

https://www.cdc.gov/coronavirus/2019-ncov/lab/resources/antigen-tests-guidelines.html#table2

There are lots of these statistics….

Table 1. Summary of statistical performance measures and their definitions used in reporting test results.

Test Classification	Reference Classification		Total	Performance Measures
	Positive	Negative		
Positive	True Positive **TP**	False Positive **FP**	All Positive Test Cases **TP + FP**	Positive Predictive Value (PPV) **TP / (TP + FP)**
Negative	False Negative **FN**	True Negative **TN**	All Negative Test Cases **FN + TN**	Negative Predictive Value (NPV) **TN / (FN + TN)**
Total	All Positive Cases **TP + FN**	All Negative Cases **FP + TN**	All Cases **TP+FN+FP+TN**	Overall Accuracy **(TP+TN) / (TP+FN+FP+TN)**
			Prevalence **(TP+FN) / (TP+FN+FP+TN)**	
Performance Measures	Sensitivity (Se) **TP / (TP + FN)**	False Positive Rate = 1 – Specificity **FP / (FP + TN)**	Likelihood Ratio Positive (LR+) **Sensitivity / (1 – Specificity)**	
	False Negative Rate = 1 – Sensitivity **FN / (TP + FN)**	Specificity (Sp) **TN / (FP + TN)**	Likelihood Ratio Negative (LR–) **(1 – Sensitivity) / Specificity**	

Proposed New Requirements for Testing and Reporting Performance Results of Arrhythmia Detection Algorithms

What does this mean for you?

Heart beats are not rare in these datasets. We want a balance of True and Negative results.

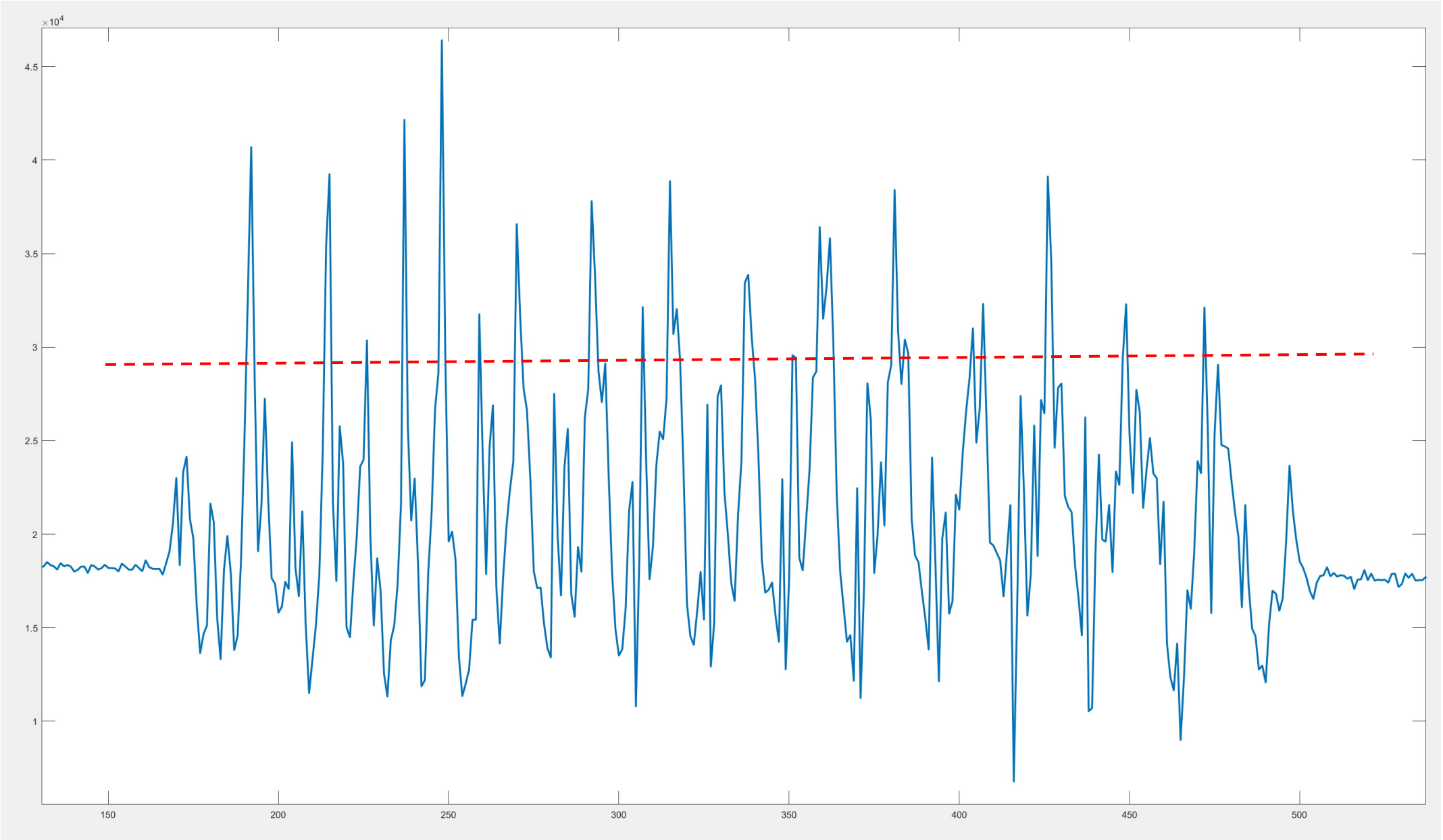
We'll focus on total measures like Accuracy and F1.

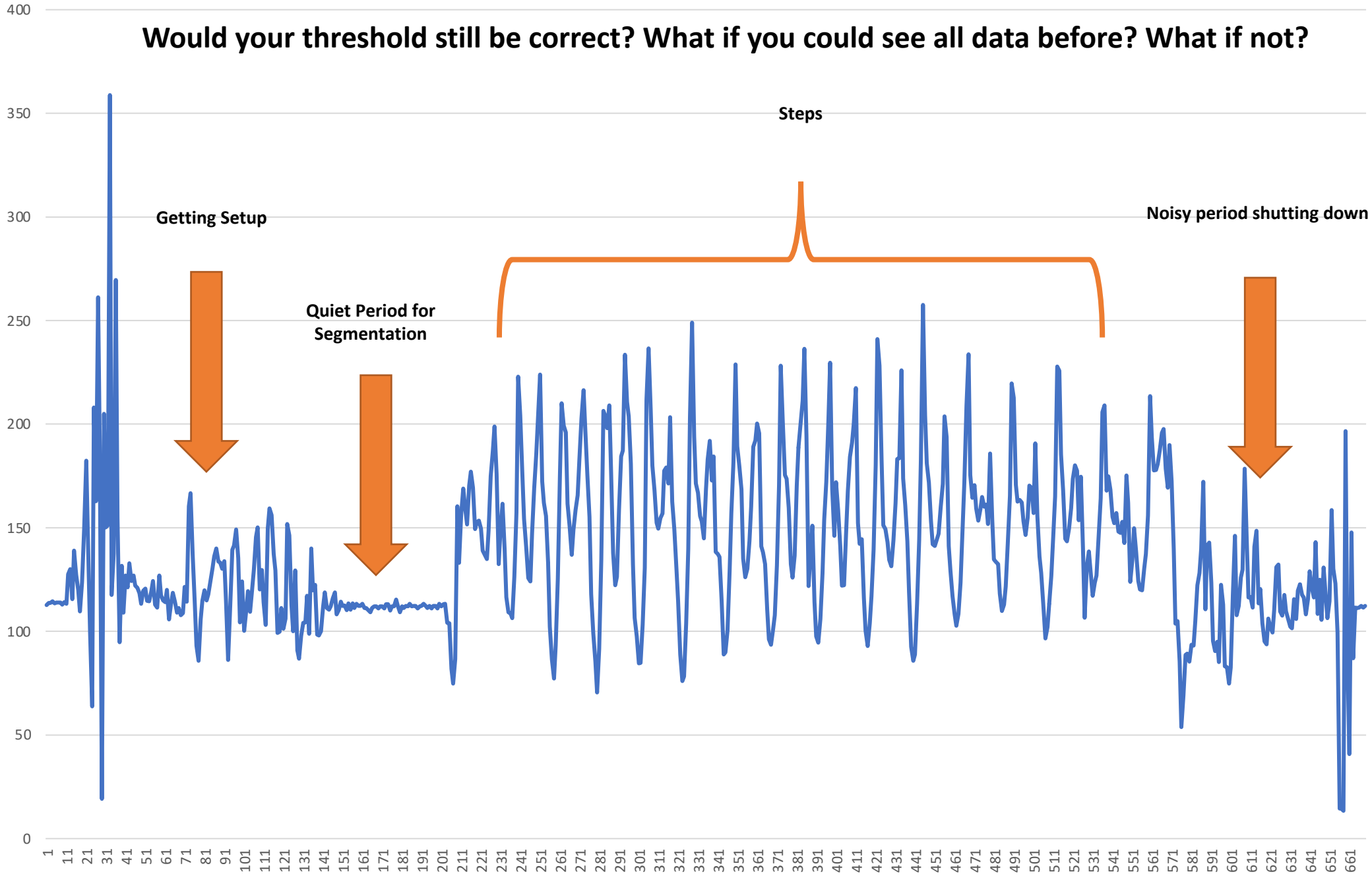
$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

$$F_1 = \frac{TP}{TP+\frac{1}{2}(FP+FN)}$$

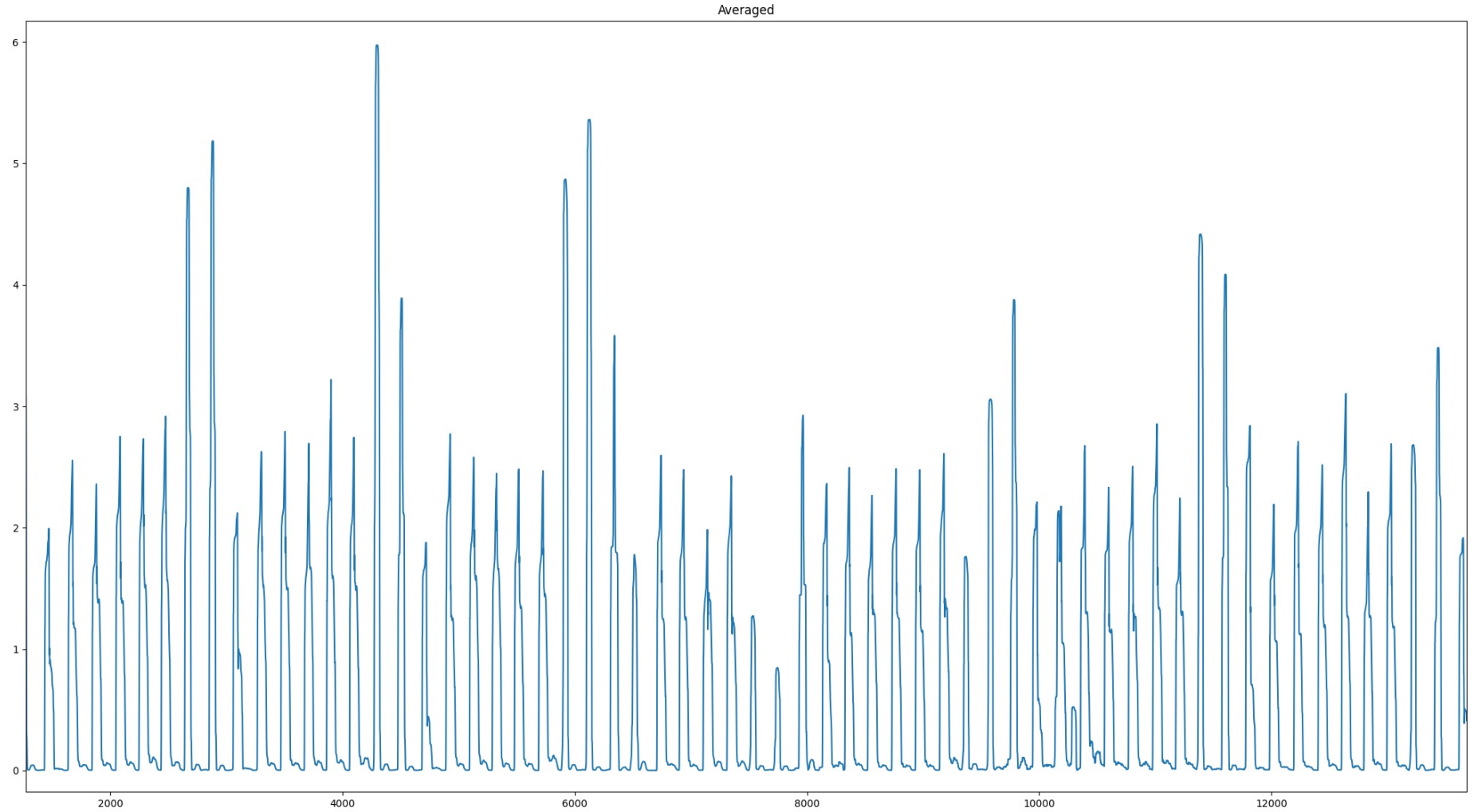
How detection works in practice.
Data can be noisy.

How many steps are in this accelerometer data? What threshold value would you pick to find them all?

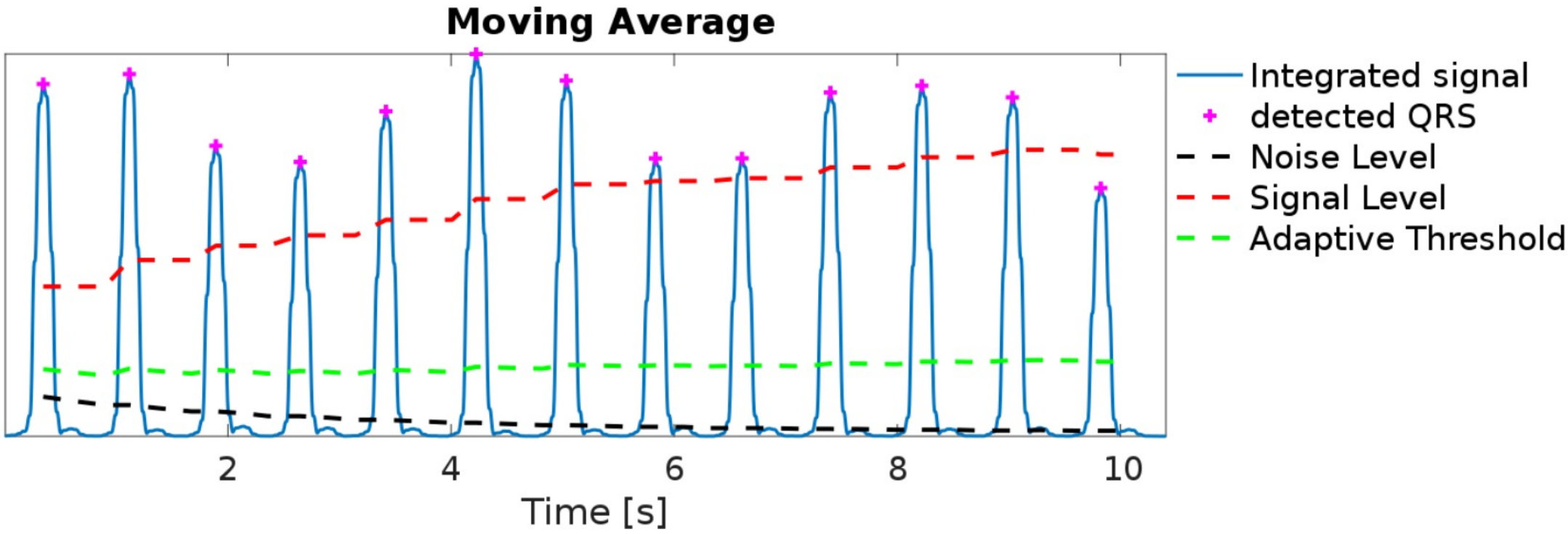



Would your threshold still be correct? What if you could see all data before? What if not?
Steps
Getting Setup
Quiet Period for Segmentation
Noisy period shutting down
400
350
300
250
200
150
100
50
0
1
11
21
31
41
51
61
71
81
91
101
111
121
131
141
151
161
171
181
191
201
211
221
231
241
251
261
271
281
291
301
311
321
331
341
351
361
371
381
391
401
411
421
431
441
451
461
471
481
491
501
511
521
531
541
551
561
571
581
591
601
611
621
631
641
651
661

Will a single static threshold work?



PT Algorithm Utilizes Adaptive Thresholds



https://commons.wikimedia.org/wiki/File:ECGrita.svg#/media/File:ECGrita.svg

How the detection happens. Online versus Offline

In the ER, is there time to record 60 minutes of data? Are there situations where this appropriate?

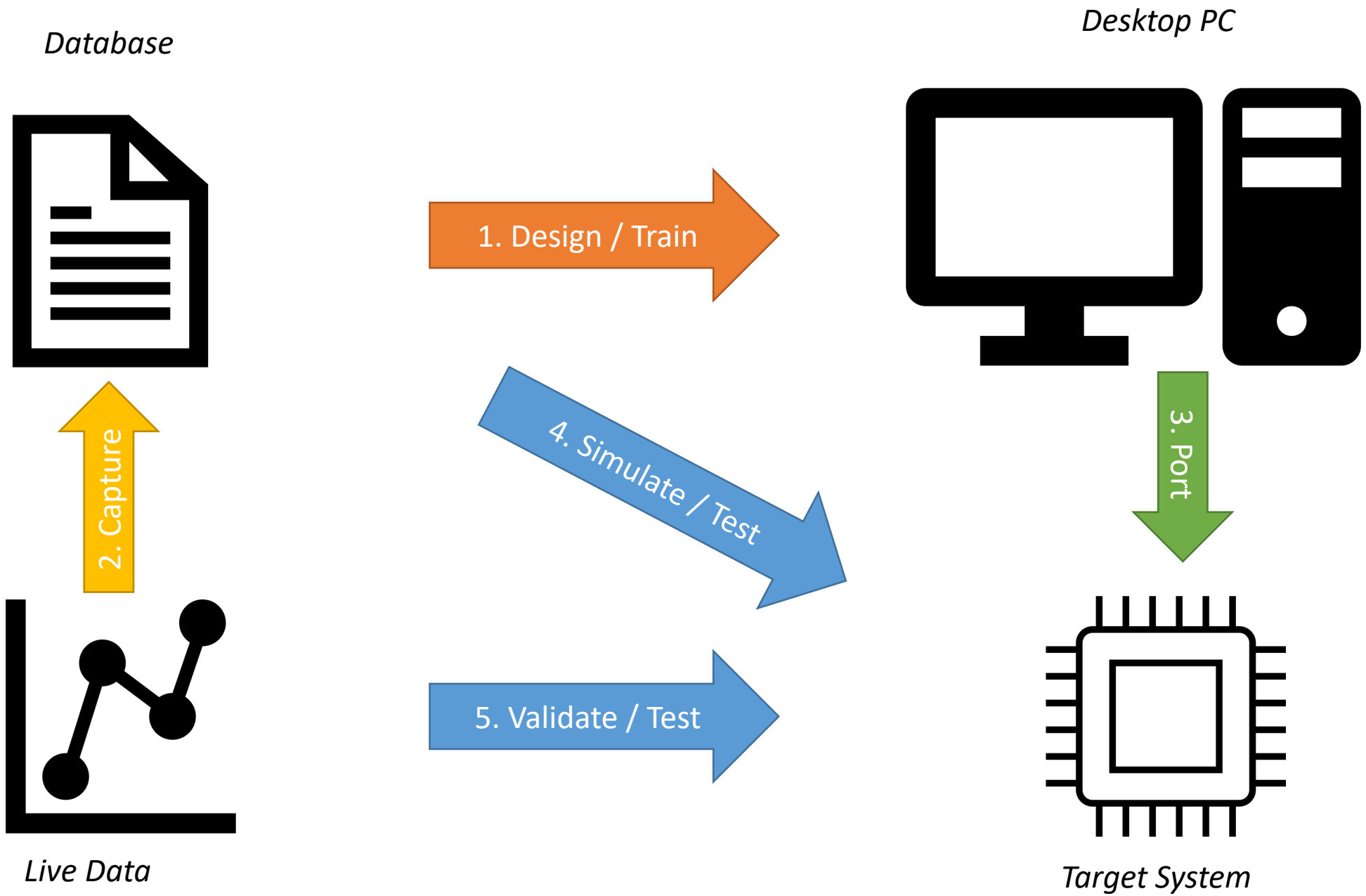
A brief digression into hardware-in-the-loop and embedded systems.

Online versus Offline Testing

- **Offline**: all the data can be accessed at once and it visible; scan and classify the events
  - Algorithm can be tuned for “correct” parameters; analysis can take “a while”
- **Online**: some data is visible, but it comes it a few points at time; a decision must be rendered within a time window (real-time deadline)
  - Algorithm can pull from trained information but must be adaptive to the current situation; may be require more “efficiency” than offline methods
- Training occurs offline; operation and validation occurs online.

Memory and CPU constrained
Decision should be rendered “soon”
Will not be running a CNN to find peaks
Will not be running numpy, scipy, pandas….

What memory or CPU constraints?
Can take its time. Mine BTC in its sleep.
Runs CNNs and CUDA for breakfast
Already has Python4 installed and found the imposter


Database
Desktop PC
1. Design / Train
2. Capture
3. Port
4. Simulate / Test
5. Validate / Test
Live Data
Target System

The implementation of your approach matters…

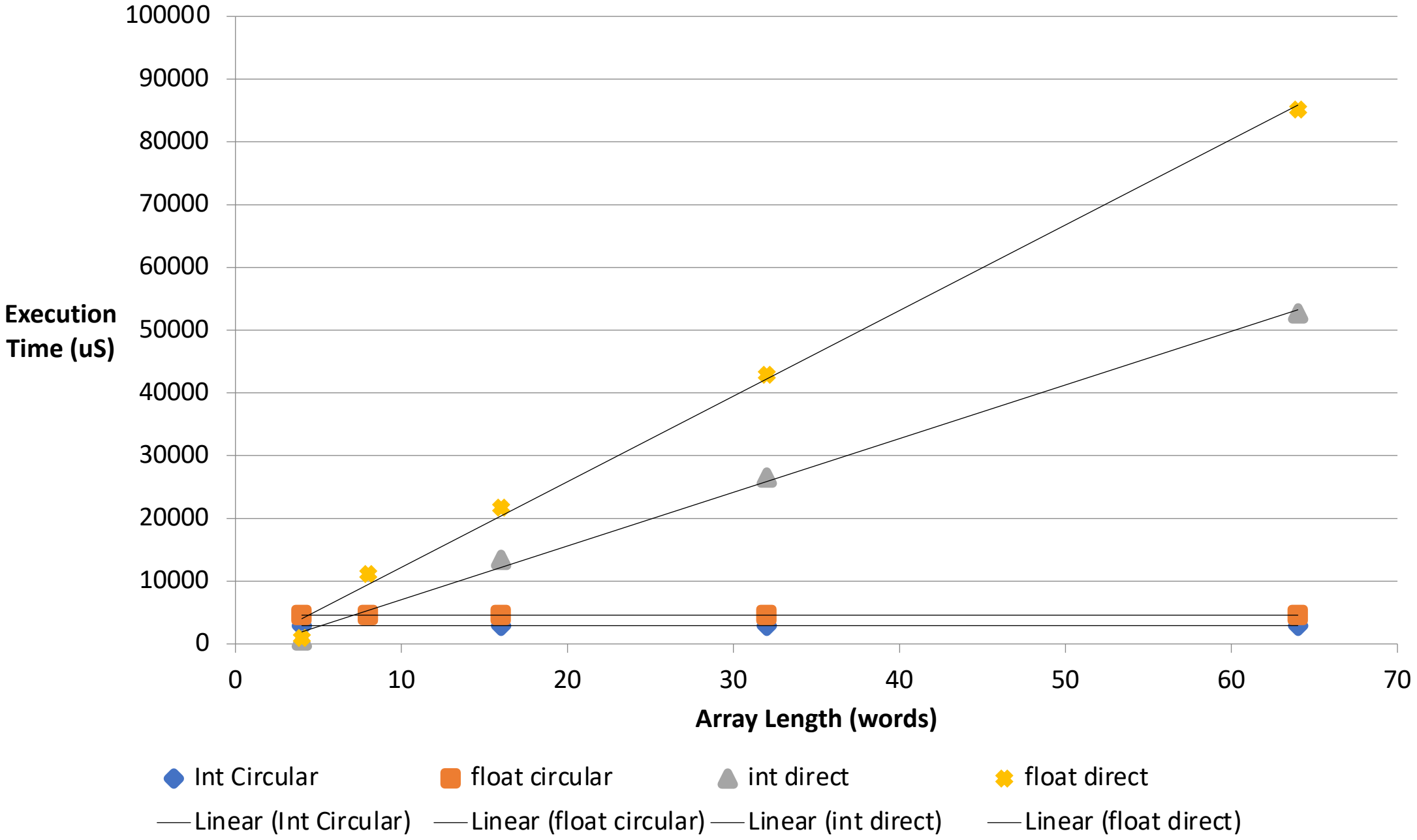
```
TYPE sum=0;
long startTime = micros();
for (int i = 0; i < LOOPS; i++)
{
  //move the data around to do the calculation
  sum = -2 * data[0] + data[1] + 4 * data[2];

  //shift the data down
  for(int j=0;j<LEN-1;j++)
  {
    data[j] = data[j+1];
  }
}
long endTime = micros();

long delta = endTime - startTime;
```

```
int counter = 0;
sum = 0;
startTime = micros();
for (int i = 0; i < LOOPS; i++)
{
  //move the calculation around on the data
  sum = -2 * data[counter % LEN] + data[(counter + 1) % LEN]
        + 4 * data[(counter + 2) % LEN];
  data[counter%LEN] = 0x37; //add some new data
  counter++;
}
endTime = micros();

delta = endTime - startTime;
```


100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
Execution Time (uS)
0
10
20
30
40
50
60
70
Array Length (words)
Int Circular
float circular
int direct
float direct
Linear (Int Circular)
Linear (float circular)
Linear (int direct)
Linear (float direct)

How this applies to us…

- Our solutions will be offline. All data will be provided and can be analyzed. A good solution will work just as well online.
- Strive for overall/balanced accuracy. Future implementations may target specific features.
- Will utilize EKG annotations to “test out” our algorithm performance.

EKG Annotations: Figuring Out if We’re Correct…

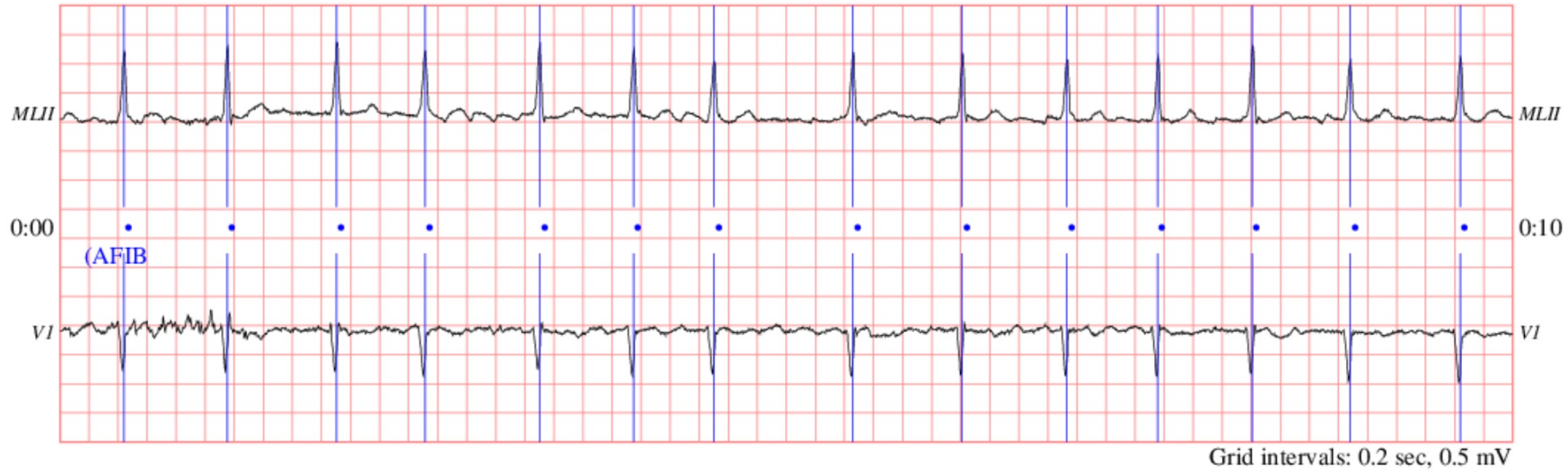
Most PhysioBank databases include one or more sets of *annotations* for each recording. Annotations are labels that point to specific locations within a recording and describe events at those locations. For example, many of the recordings that contain ECG signals have annotations that indicate the times of occurrence and types of each individual heart beat ("beat-by-beat annotations").

Each instance of an annotation may have up to six attributes:

- `time`: the time within the recording (recorded in the annotation file as the sample number of the sample to which the annotation "points")
- `anntyp` [sic]: a numeric annotation code (see `ecgcodes.h` for definitions)
- `subtyp` [sic], `chan`, `num`: three small integers (between -128 to 127) that specify context-dependent attributes (see the documentation for each database for details)
- `aux`: a free text string

https://archive.physionet.org/physiobank/annotations.shtml

How these annotations look online…



Symbols used in plots

[An expanded and updated version of the table below can

Symbol	
· *or* N	Normal beat
L	Left bundle branch block beat
R	Right bundle branch block beat
A	Atrial premature beat
a	Aberrated atrial premature beat
J	Nodal (junctional) premature beat
S	Supraventricular premature beat
V	Premature ventricular contraction
F	Fusion of ventricular and normal beat
[	Start of ventricular flutter/fibrillation
!	Ventricular flutter wave
]	End of ventricular flutter/fibrillation
e	Atrial escape beat
j	Nodal (junctional) escape beat
E	Ventricular escape beat
/	Paced beat
f	Fusion of paced and normal beat
x	Non-conducted P-wave (blocked APB)
Q	Unclassifiable beat
|	Isolated QRS-like artifact

Use the PhysioBank ATM: https://archive.physionet.org/cgi-bin/atm/ATM

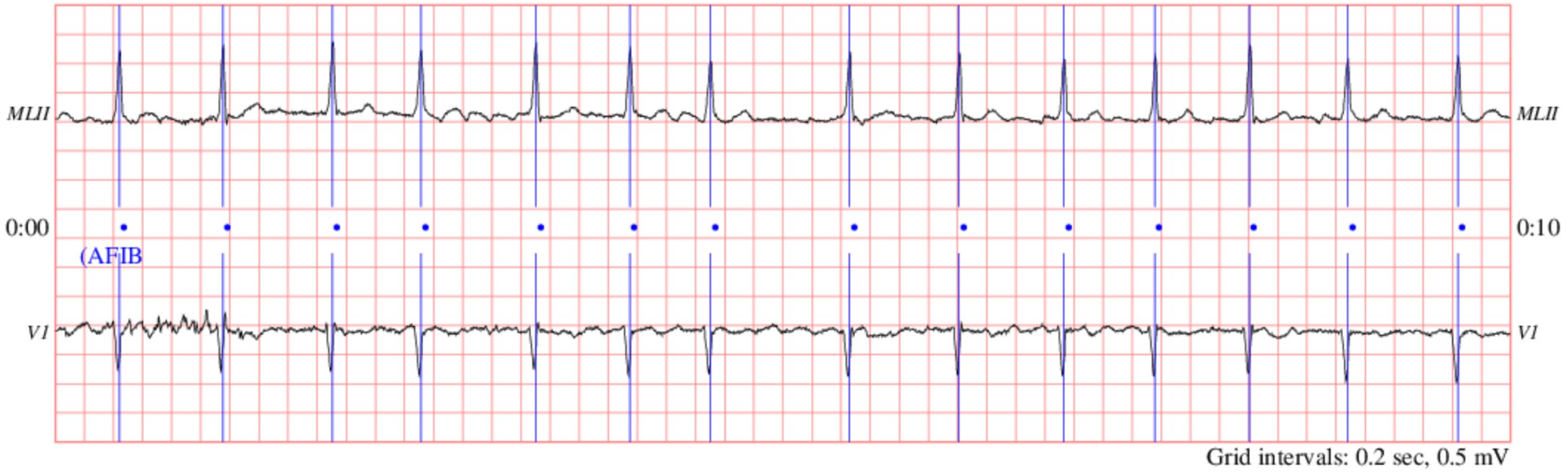
MITDB: https://archive.physionet.org/physiobank/database/html/mitdbdir/intro.htm

How these annotations look in the annotations file…

0:00.167	60	+	0	0	0	(AFIB
0:00.442	159	N	0	0	0	
0:01.153	415	N	0	0	0	
0:01.906	686	N	0	0	0	
0:02.514	905	N	0	0	0	
0:03.303	1189	N	0	0	0	
0:03.950	1422	N	0	0	0	
0:04.503	1621	N	0	0	0	
0:05.458	1965	N	0	0	0	
0:06.211	2236	N	0	0	0	
0:06.931	2495	N	0	0	0	
0:07.558	2721	N	0	0	0	
0:08.208	2955	N	0	0	0	
0:08.881	3197	N	0	0	0	
0:09.639	3470	N	0	0	0	

Code	Description
N	Normal beat (displayed as "·" by the PhysioBank ATM, Light\
L	Left bundle branch block beat
R	Right bundle branch block beat
B	Bundle branch block beat (unspecified)
A	Atrial premature beat
a	Aberrated atrial premature beat
J	Nodal (junctional) premature beat
S	Supraventricular premature or ectopic beat (atrial or nodal)
V	Premature ventricular contraction
r	R-on-T premature ventricular contraction
F	Fusion of ventricular and normal beat
e	Atrial escape beat
j	Nodal (junctional) escape beat
n	Supraventricular escape beat (atrial or nodal)
E	Ventricular escape beat
/	Paced beat
f	Fusion of paced and normal beat
Q	Unclassifiable beat
?	Beat not classified during learning

Time	Sample	Annot.				Rhythm Annot.
0:00.167	60	+	0	0	0	(AFIB
0:00.442	159	N	0	0	0	
0:01.153	415	N	0	0	0	
0:01.906	686	N	0	0	0	
			Ch.	Num.	Extra	



Rhythm Annotations

2. In rhythm, ST segment, and T-wave change annotations, and in measurement and comment annotations, the `aux` field contains an ASCII string (with prefixed byte count) describing the rhythm, ST segment, T-wave change, measurement, or the nature of the comment. By convention, the character that follows the byte count in the `aux` field of a `+` annotation is "`(`". The most commonly used rhythm annotation strings are:

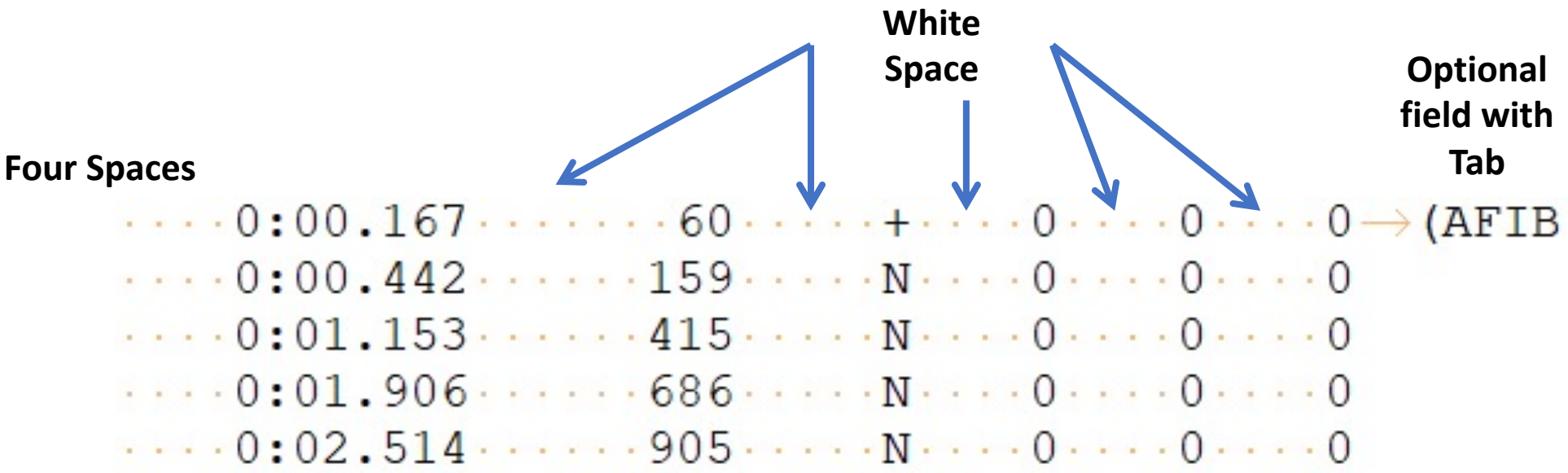
String	Description
(AB	Atrial bigeminy
(AFIB	Atrial fibrillation
(AFL	Atrial flutter
(B	Ventricular bigeminy
(BII	2° heart block
(IVR	Idioventricular rhythm
(N	Normal sinus rhythm
(NOD	Nodal (A-V junctional) rhythm
(P	Paced rhythm
(PREX	Pre-excitation (WPW)
(SBR	Sinus bradycardia
(SVTA	Supraventricular tachyarrhythmia
(T	Ventricular trigeminy
(VFL	Ventricular flutter
(VT	Ventricular tachycardia

https://archive.physionet.org/physiobank/annotations.shtml#aux

Databases may follow different conventions. Double check before reading in annotations.

We will attempt to use the most regular annotation structure.

Parsing Annotation Files…



Gross… I will provide parser to deal with annotation files. Or, you can use wfdb package. Your choice.