data sniffing

aka initial data analysis

Fonti Kar + Emi Tanaka

GRAIN TRADE AUSTRALIA

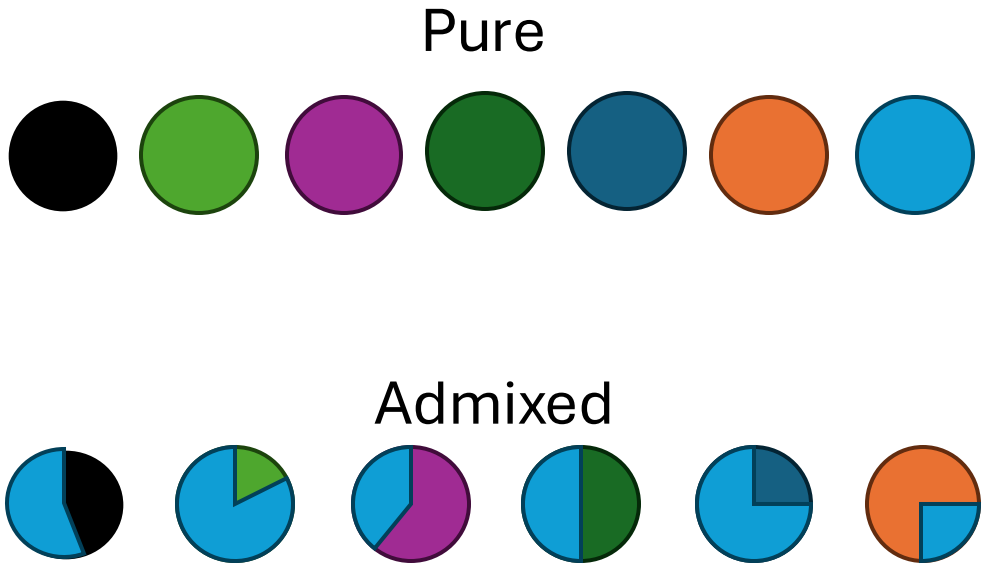
ZOOMAGRI

- Long term storage of harvest from multiple farmers
- Need solution to identify / verify grains

- Computer vision / machine learning tech to grade barley
- Device trained for 7* varieties

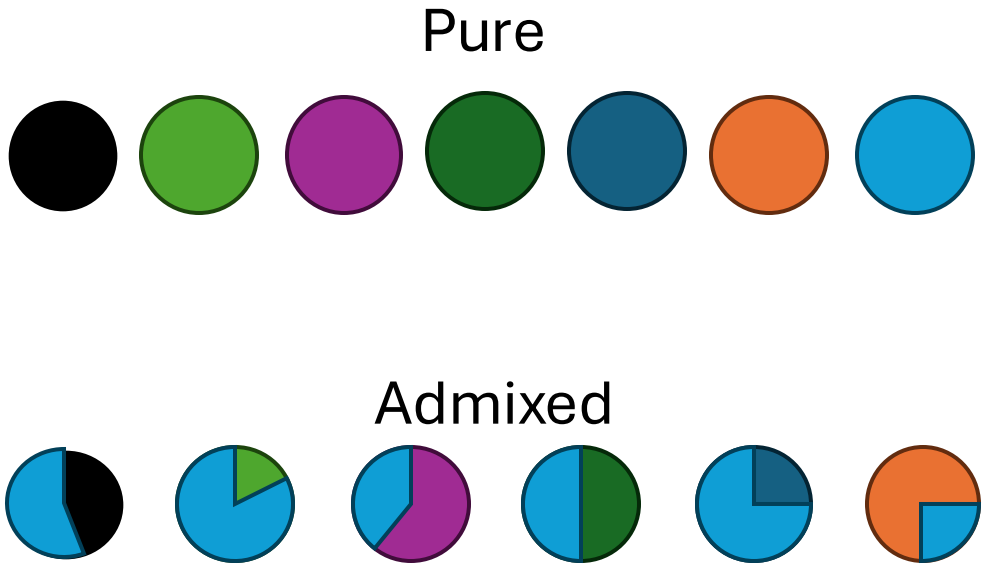
methods

- 10 devices
- 7 barley varietals
- 24 samples
  - 12 x Pure
  - 12 x Admixed



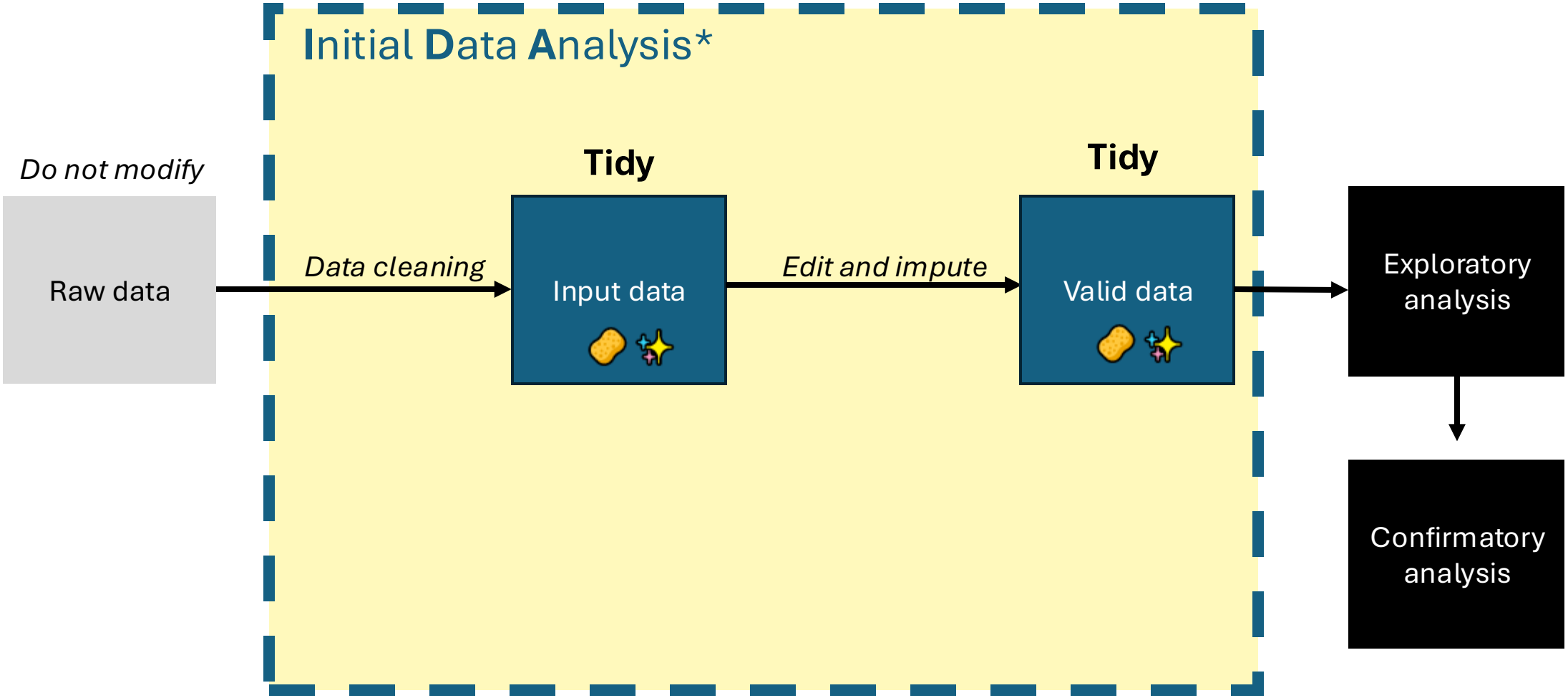
methods

- 10 devices
- 7 barley varietals
- 24 samples
  - 12 x Pure
  - 12 x Admixed



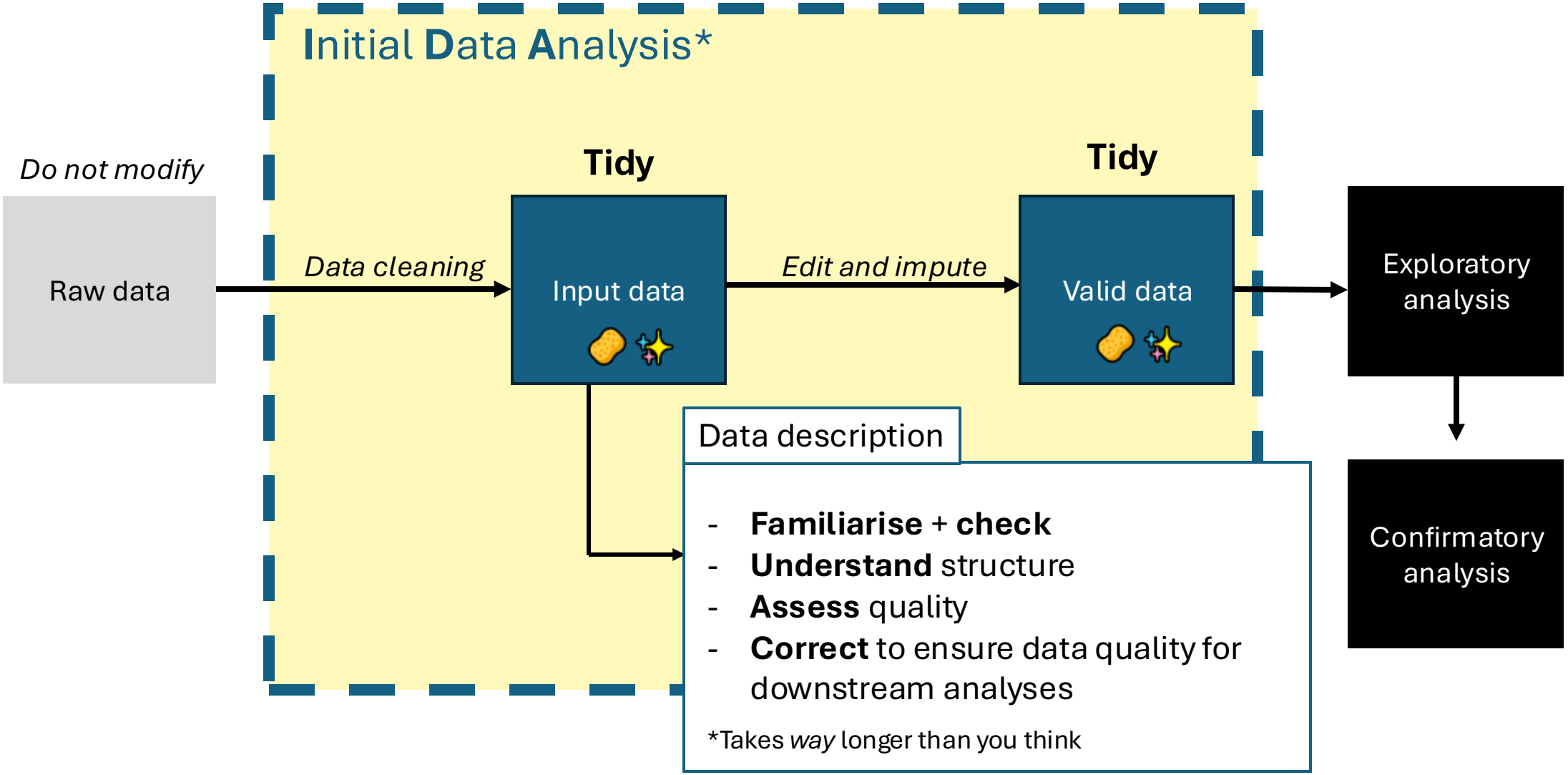
What is the **efficacy** and **consistency** of the ZoomAgri devices in detecting the correct barley varietal(s)?

data analysis pipeline



Huebner et al (2018) A Contemporary Conceptual Framework for Initial Data Analysis. Observational Studies
van der Loo & de Jonge (2018) Statistical Data Cleaning with Applications in R.

data analysis pipeline



🥔 tidy data ✨

Why tidy?

Complex data analyses and tools uses **programming languages that require tidiness**

Date	10-Jun-2017		Notes
Recorder	Spongebob Squarepants		Plant 3 was on a different shelf as 2 and 1
	Plant 1	Plant 2	Plant 3
Treatment a	2.9 (18%)	3.6 (25%)	
Treatment b			1.7 (40%)

- Human-friendly for data collection

Wickham (2014) Tidy Data. *Journal of Statistical Software*

🥔 tidy data ✨

✓Each column is a variable.
✓Each row is an observation.
✓Each cell is a single value.

date	recorder	subject	treatment	leaf area	infected percent
10-06-2017	SB	plant 1	a	2.9	18
10-06-2017	SB	plant 2	a	3.6	25
10-06-2017	SB	plant 3	b	1.7	40

- Condensed
- Each cell has a single purpose

Wickham (2014) Tidy Data. *Journal of Statistical Software*

🥔 tidy data ✨

✓Each column is a variable.
✓Each row is an observation.
✓Each cell is a single value.

Let's have a go! 🤓 **(5 mins)**

- What makes these spreadsheets untidy?
- What would their tidy version look like?

Spreadsheet 1

	A	B	C	D	E	F	G	H	I
1		1 min				5 min			
2	strain	normal		mutant		normal		mutant	
3	A	147	139	166	179	334	354	451	474
4	B	246	240	178	172	514	611	412	447

	A	B	C	D	E	F	G	H	I
1		1 min				5 min			
2	strain	normal		mutant		normal		mutant	
3	A	147	139	166	179	334	354	451	474
4	B	246	240	178	172	514	611	412	447

if in doubt, count it out

Count your key variables to:

- **verify + understand** structure
- identify **missing / strange** values

```
raw_data |>
  count(workstation)

raw_data |>
  count(sample_code)
```

```
> raw_data |>
+   count(workstation)
# A tibble: 11 x 2
   workstation        n
   <chr>          <int>
 1 Beulah            30
 2 Deniliquin        30
 3 Donald Bunker     30
 4 Dunolly           30
 5 Manangatang       30
 6 Murtoa South      30
 7 Piangil           30
 8 Quambatook        33
 9 Ultima            30
10 Warracknabeal     30
11 NA                 2
```

INPUT

```
> raw_data |>
+   count(sample_code) |>
+   print(n = 50)
# A tibble: 38 x 2
   sample_code              n
   <chr>                <int>
 1 GTABottler              10
 2 GTACyclops              10
 3 GTAMinotaur              9
 4 GTAMix1                 10
 5 GTAMix10                10
 6 GTAMix11                10
 7 GTAMix12                10
 8 GTAMix2                 10
 9 GTAMix3                 10
10 GTAMix4                 10
11 GTAMix5                 10
12 GTAMix6                 10
13 GTAMix7                 10
14 GTAMix8                 10
15 GTAMix9                 10
16 GTANeo                  10
17 GTAPure1                10
18 GTAPure10               10
19 GTAPure10 - Repeat1      1
20 GTAPure10 - Repeat2      1
21 GTAPure10 - Repeat3      1
22 GTAPure11               10
23 GTAPure12               10
24 GTAPure2                10
25 GTAPure3                10
26 GTAPure4                10
27 GTAPure5                10
28 GTAPure6                10
29 GTAPure7                10
30 GTAPure8                10
31 GTAPure9                10
32 GTATItan                 1
33 GTATitan                 8
34 GTAZena                  9
35 Mino - not done          1
36 TItan - not done         1
37 Zena - not done          1
38 NA                       2
```

INPUT

if in doubt, count it out

Count your key variables to:

- **verify + understand** structure
- identify **missing / strange** values
- identify the **observational unit**

Observational unit is smallest entity from which data has been collected from

```
raw_data |>
tabyl(workstation, sample_code)
```

- Determines *independence* and *unit of analysis*

```
> raw_data |>
+   tabyl(workstation, sample_code)
   workstation GTAMix1 GTAMix10 GTAMix11 GTAMix12 GTAMix2 GTAMix3 GTAMix4 GTAMix5 GTAMix6 GTAMix7 GTAMix8 GTAMix9
        Beulah       1        1        1        1       1       1       1       1       1       1       1       1
    Deniliquin       1        1        1        1       1       1       1       1       1       1       1       1
 Donald Bunker       1        1        1        1       1       1       1       1       1       1       1       1
       Dunolly       1        1        1        1       1       1       1       1       1       1       1       1
   Manangatang       1        1        1        1       1       1       1       1       1       1       1       1
  Murtoa South       1        1        1        1       1       1       1       1       1       1       1       1
       Piangil       1        1        1        1       1       1       1       1       1       1       1       1
    Quambatook       1        1        1        1       1       1       1       1       1       1       1       1
        Ultima       1        1        1        1       1       1       1       1       1       1       1       1
 Warracknabeal       1        1        1        1       1       1       1       1       1       1       1       1
 GTAPure1 GTAPure10 GTAPure11 GTAPure12 GTAPure2 GTAPure3 GTAPure4 GTAPure5 GTAPure6 GTAPure7 GTAPure8 GTAPure9
        1         1         1         1        1        1        1        1        1        1        1        1
        1         1         1         1        1        1        1        1        1        1        1        1
        1         1         1         1        1        1        1        1        1        1        1        1
        1         1         1         1        1        1        1        1        1        1        1        1
        1         1         1         1        1        1        1        1        1        1        1        1
        1         1         1         1        1        1        1        1        1        1        1        1
        1         1         1         1        1        1        1        1        1        1        1        1
        1         4         1         1        1        1        1        1        1        1        1        1
        1         1         1         1        1        1        1        1        1        1        1        1
        1         1         1         1        1        1        1        1        1        1        1        1
```

INPUT

don’t be afraid to go digging ⍰

- What are **classified seeds**? 🧐
- **Supplement** your understanding with more info

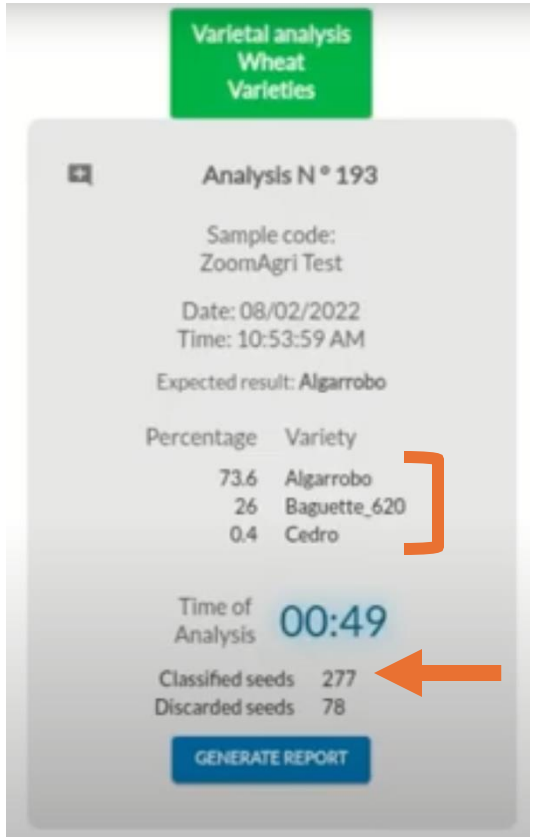
workstation	analysis_id	sample_code	classified_seeds	result_1	result_1_percent	result_2	result_2_percent	result_3	result_3_percent
Beulah	200	GTAMix1	303	Bass	84	Maximus CL	14	Others	2
Beulah	208	GTAMix10	283	Maximus CL	65	Bass	35	NA	NA
Beulah	209	GTAMix11	312	La Trobe	50	Maximus CL	50	NA	NA
Beulah	210	GTAMix12	273	Planet	54	Spartacus CL	46	NA	NA
Beulah	223	GTAMix2	343	Planet	84	Bass	16	NA	NA

INPUT

don’t be afraid to go digging 🔍

- What are **classified seeds**? 🧐
- **Supplement** your understanding with more info

Number of seeds is easier to model than percentages 🍕



INPUT

drop dead weight

Discard variables that are :

- **redundant**
- **don't contribute** information

```
raw_data |>
  janitor::remove_constant()
```

```
Removing 2 constant columns of 16 columns total (Removed: number_samples, expected_result).
```

```
raw_data |>
  janitor::get_one_to_one()
```

```
> raw_data |>
+   janitor::get_one_to_one()
[[1]]
[1] "workstation" "pc_id"       "s_n"
```

```
raw_data |>
  count(number_samples, expected_result)
```

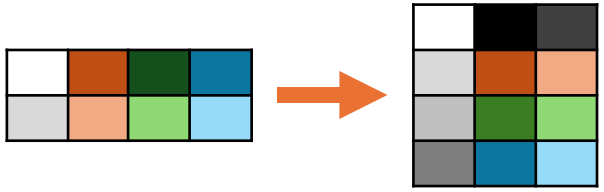
```
> raw_data |>
+   count(number_samples, expected_result)
# A tibble: 1 x 3
  number_samples expected_result     n
           <dbl> <chr>           <int>
1              1 Unknown           243
```

INPUT

take + make what you need

- **Create, retain, reformat** variables that are needed downstream
  - Make it 🥔 tidy ✨

```
raw_data |>
  pivot_longer()
```



wide

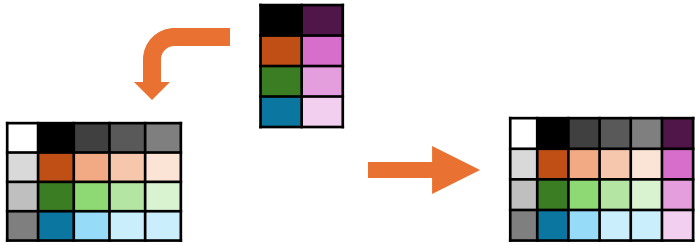
workstation	analysis_id	sample_code	classified_seeds	result_1	result_1_percent	result_2	result_2_percent	result_3	result_3_percent
Beulah	200	GTAMix1	303	Bass	84	Maximus CL	14	Others	2
Beulah	208	GTAMix10	283	Maximus CL	65	Bass	35	NA	NA
Beulah	209	GTAMix11	312	La Trobe	50	Maximus CL	50	NA	NA
Beulah	210	GTAMix12	273	Planet	54	Spartacus CL	46	NA	NA
Beulah	223	GTAMix2	343	Planet	84	Bass	16	NA	NA

long

workstation	analysis_id	sample_code	classified_seeds	obs_id	variety	percent	result_top
Beulah	200	GTAMix1	303	1	Bass	84	1
Beulah	200	GTAMix1	303	1	Maximus CL	14	2
Beulah	200	GTAMix1	303	1	Others	2	3
Beulah	208	GTAMix10	283	2	Maximus CL	65	1
Beulah	208	GTAMix10	283	2	Bass	35	2
Beulah	209	GTAMix11	312	3	La Trobe	50	1
Beulah	209	GTAMix11	312	3	Maximus CL	50	2

take + make what you need

- **Create, retain, reformat** variables that are needed downstream
  - e.g. expected percentages



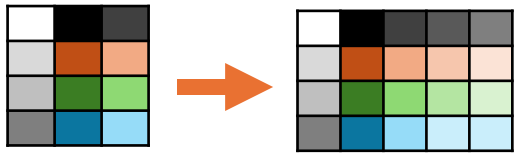
workstation	analysis_id	sample_code	classified_seeds	obs_id	variety	percent	percent_truth	result_top
Beulah	200	GTAMix1	303	1	Bass	84	85	1
Beulah	200	GTAMix1	303	1	Maximus CL	14	15	2
Beulah	200	GTAMix1	303	1	Others	2	0	3
Beulah	200	GTAMix1	303	1	Planet	0	0	NA
Beulah	200	GTAMix1	303	1	Spartacus CL	0	0	NA
Beulah	200	GTAMix1	303	1	Westminster	0	0	NA
Beulah	200	GTAMix1	303	1	La Trobe	0	0	NA
Beulah	200	GTAMix1	303	1	Commodus CL	0	0	NA
Beulah	208	GTAMix10	283	2	Maximus CL	65	65	1
Beulah	208	GTAMix10	283	2	Bass	35	35	2
		GTAMix10	283	2	Planet	0	0	NA
		GTAMix10	283	2	Spartacus CL	0	0	NA
		GTAMix10	283	2	Westminster	0	0	NA
		GTAMix10	283	2	La Trobe	0	0	NA
Beulah	208	GTAMix10	283	2	Commodus CL	0	0	NA

```
raw_data |>
  left_join(expected_percentages)
```

VALID

take + make what you need

- **Create, retain, reformat** variables that are needed downstream
  - e.g. predicted and expected seeds



```
raw_data |>
  mutate(predicted_seeds = percent/100 * classified_seeds,
         expected_seeds = percent_truth/100 * classified_seeds)
```

workstation	sample_code	classified_seeds	obs_id	variety	predicted_percent	expected_percent	result_top	predicted_seeds	expected_seeds	sample_type
Beulah	GTAMix1	303	1	Bass	84	85	1	255	258	Mix
Beulah	GTAMix1	303	1	Maximus CL	14	15	2	42	45	Mix
Beulah	GTAMix1	303	1	Others	2	0	3	6	0	Mix
Beulah	GTAMix1	303	1	Planet	0	0	NA	0	0	Mix
Beulah	GTAMix1	303	1	Spartacus CL	0	0	NA	0	0	Mix
Beulah	GTAMix1	303	1	Westminster	0	0	NA	0	0	Mix
Beulah	GTAMix1	303	1	La Trobe	0					
Beulah	GTAMix1	303	1	Commodus CL	0					
Beulah	GTAMix10	283	2	Maximus CL	65					
Beulah	GTAMix10	283	2	Bass	35					
Beulah	GTAMix10	283	2	Planet	0					
Beulah	GTAMix10	283	2	Spartacus CL	0	0	NA	0	0	Mix
Beulah	GTAMix10	283	2	Westminster	0	0	NA	0	0	Mix
Beulah	GTAMix10	283	2	La Trobe	0	0	NA	0	0	Mix
Beulah	GTAMix10	283	2	Commodus CL	0	0	NA	0	0	Mix

Number of seeds is easier to model than percentages 🎉

VALID

explore your question with visuals 🎨

GRAIN TRADE AUSTRALIA

🧑 What is the **efficacy** and **consistency** of the ZoomAgri devices in detecting the correct barley varietal(s)?

Efficacy 🎯

- Can the **devices** predict the **correct variety** / mix of varietals for each **sample?**

➞ <u>Compare</u> between 'truth' and device output

explore your question with visuals 🎨

GRAIN TRADE AUSTRALIA

🧑 What is the **efficacy** and **consistency** of the ZoomAgri devices in detecting the correct barley varietal(s)?

Efficacy 🎯

- Can the **devices** predict the **correct variety** / mix of varietals for each **sample?**

➡ Compare between 'truth' and device output

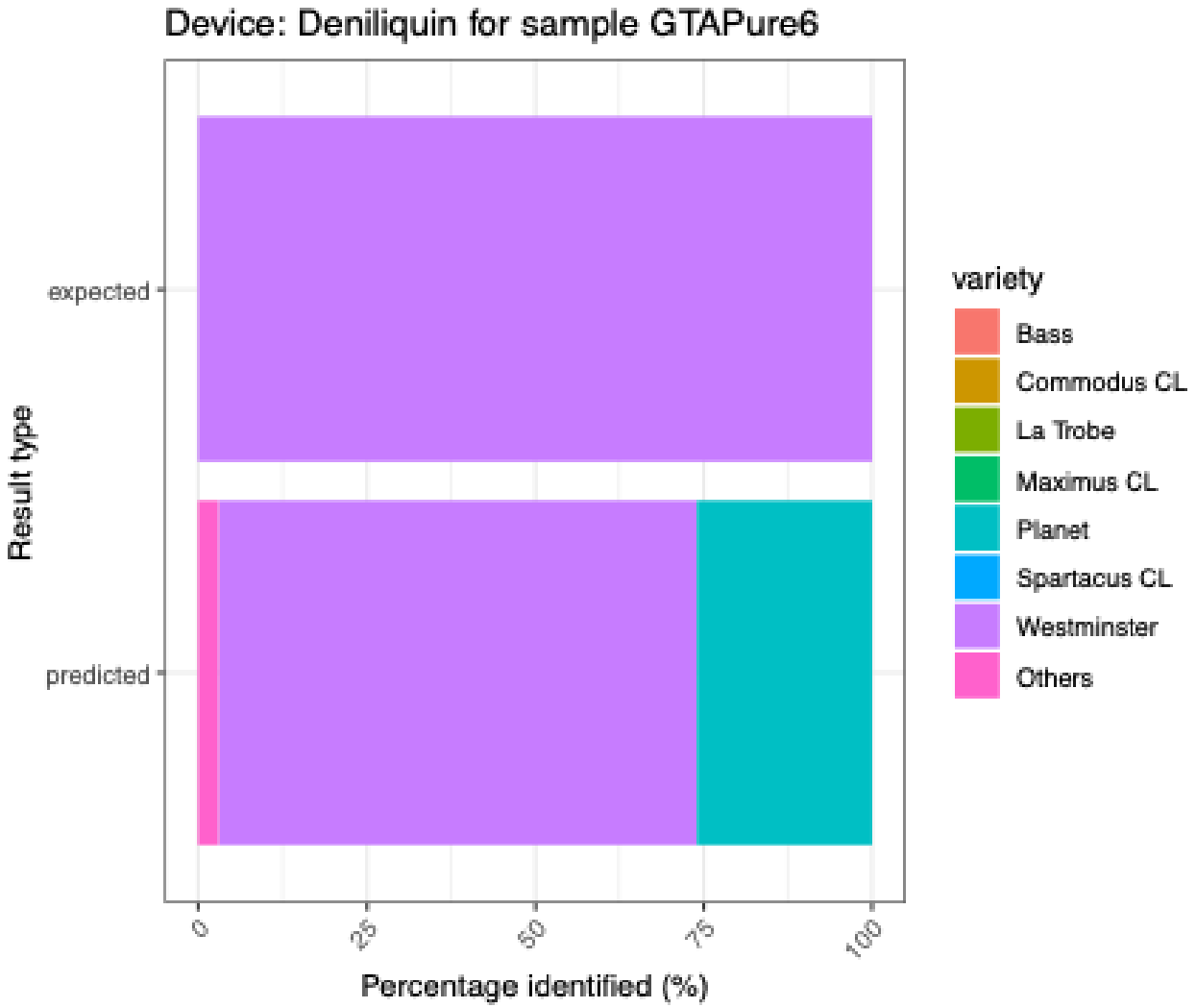
Consistency 🎯 🎯 🎯

- Are all **devices** performing **similarly?**

➡ Show agreement across all devices and samples

Efficacy 🎯: Compare between 'truth' and device output

start simple;



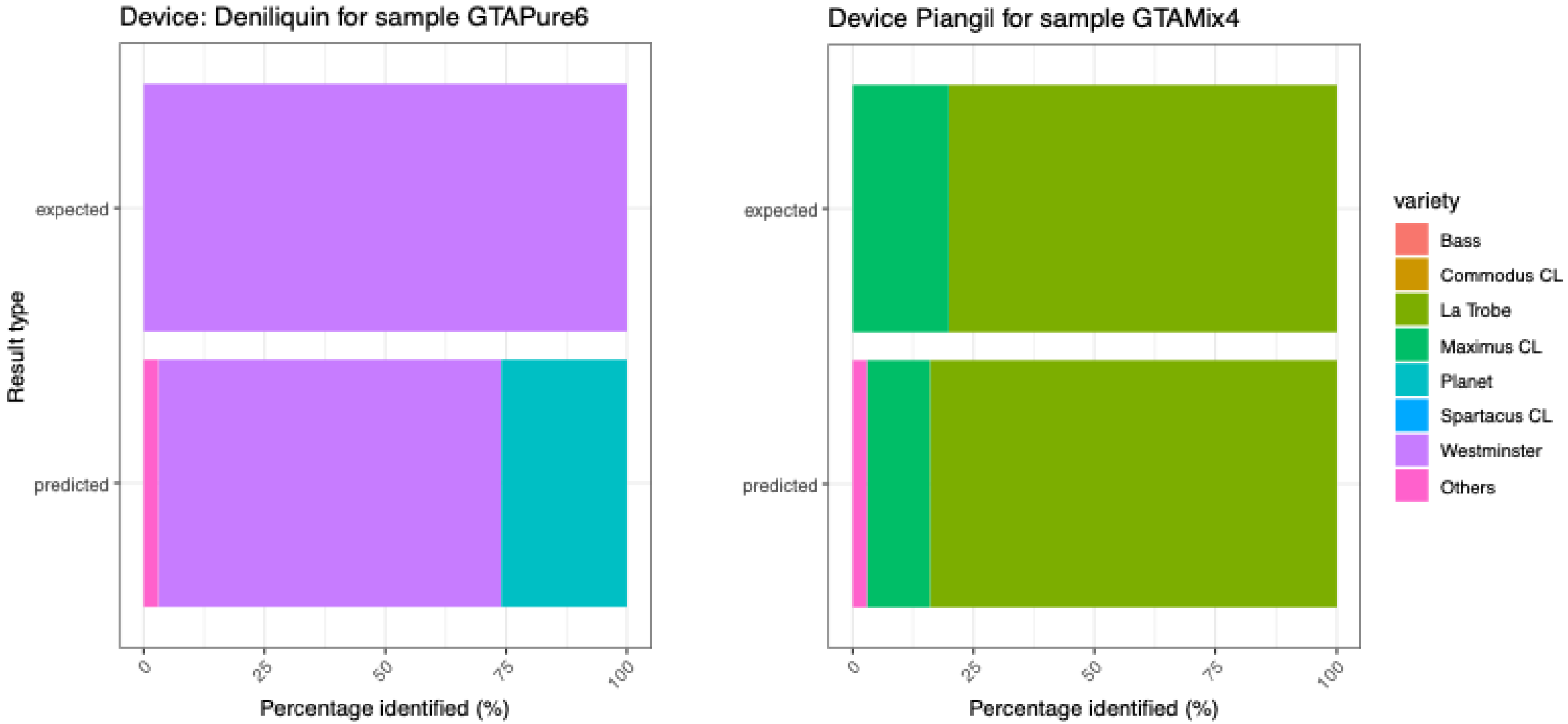
Data viz 🎨

- Top and bottom bars for **visual comparison**
- **Stacked bar shows composition** of device prediction

Cleveland, W. S., & McGill, R. (1984). *Graphical perception: Theory, experimentation, and application to the development of graphical methods.*

Efficacy 🎯: Compare between 'truth' and device output

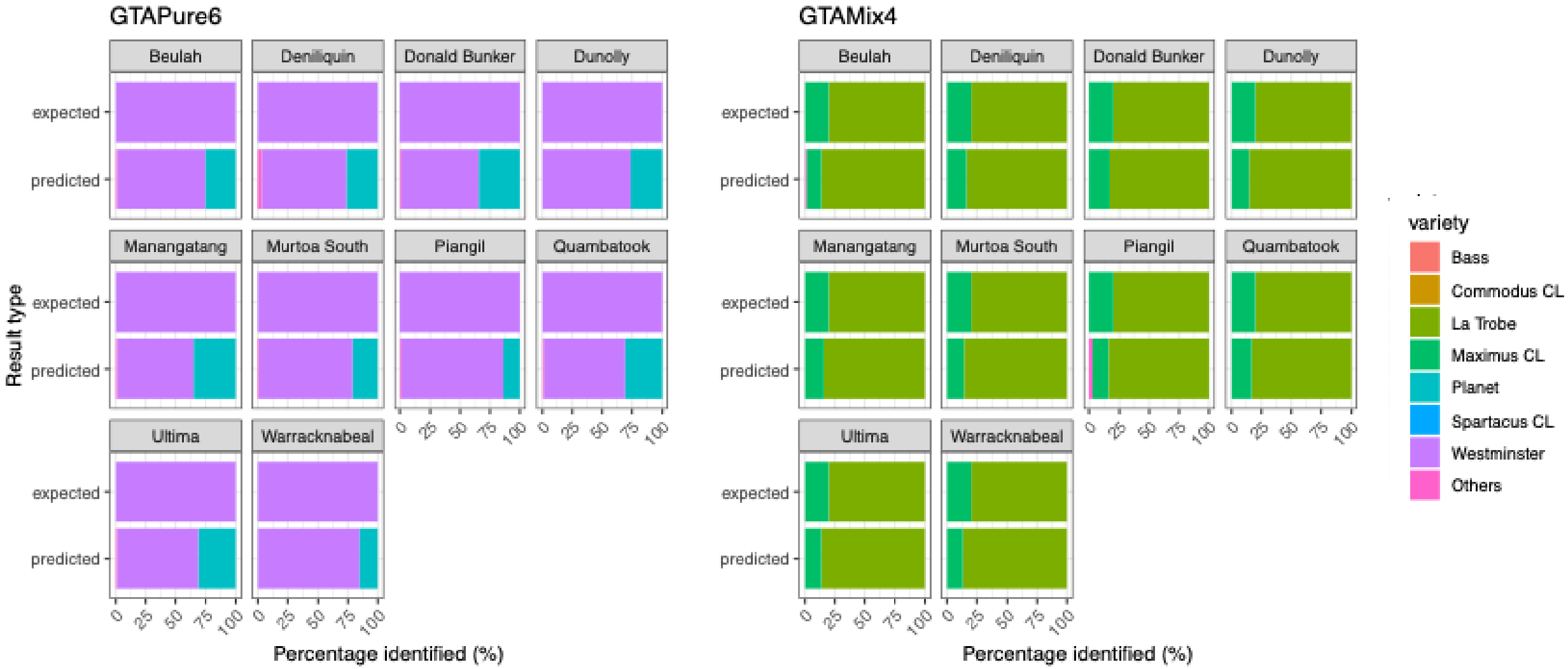
start simple;



Cleveland, W. S., & McGill, R. (1984). *Graphical perception: Theory, experimentation, and application to the development of graphical methods.*

Efficacy 🎯: Compare between 'truth' and device output

start simple; then scale up



Cleveland, W. S., & McGill, R. (1984). *Graphical perception: Theory, experimentation, and application to the development of graphical methods.*

Admixed samples

variety

- Bass
- Commodus CL
- La Trobe
- Maximus CL
- Planet
- Spartacus CL
- Westminster
- Others

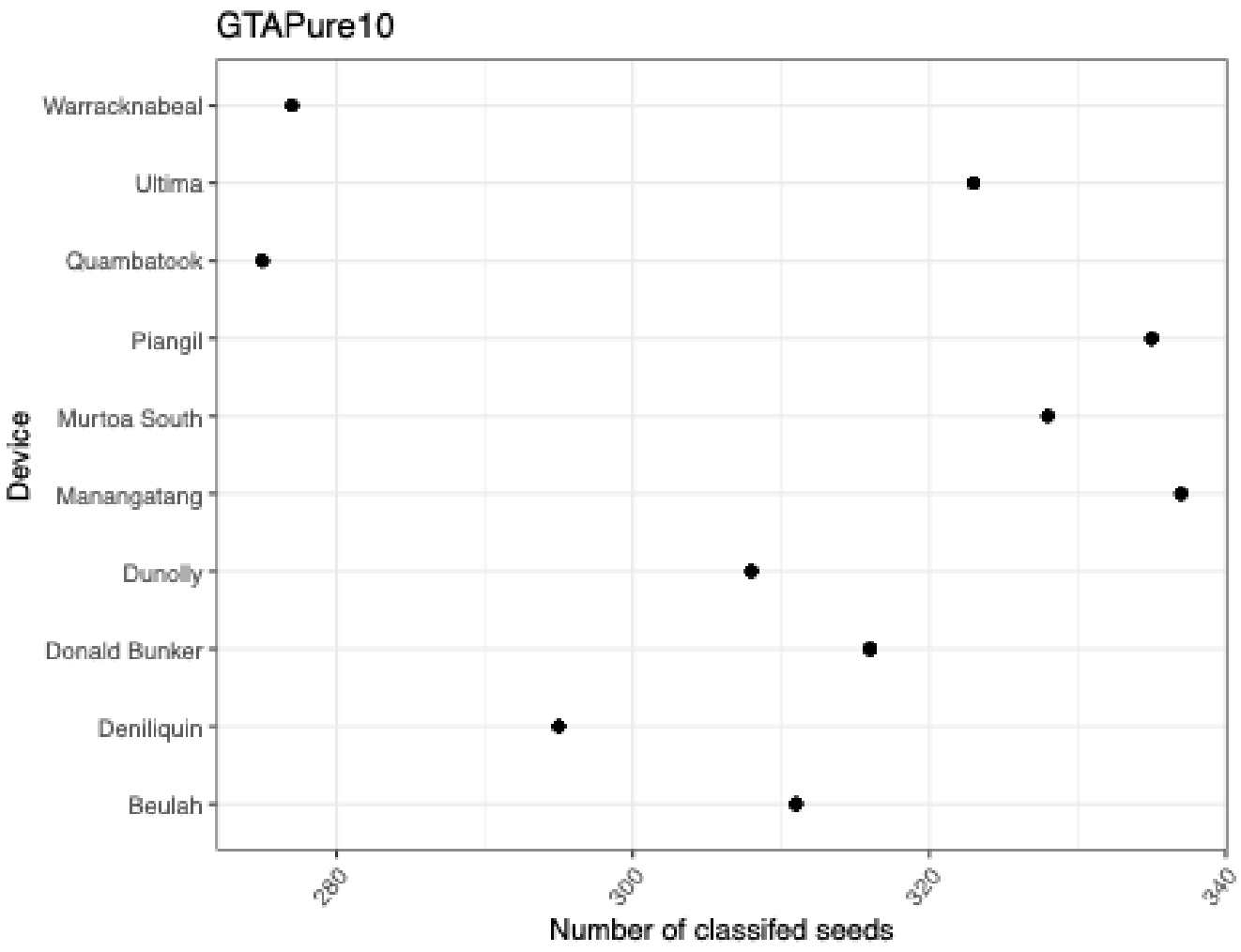
Admixed samples

variety

- Bass
- Commodus CL
- La Trobe
- Maximus CL
- Planet
- Spartacus CL
- Westminster
- Others

Consistency 🎯 : Show agreement across all devices and samples

could classified seeds tell us something about device consistency?



Data viz 🎨

- Stacking device for **visual comparison**
- **Alignment** of points implies agreement*

*Assumes same number of seeds used for every evaluation, sample, device

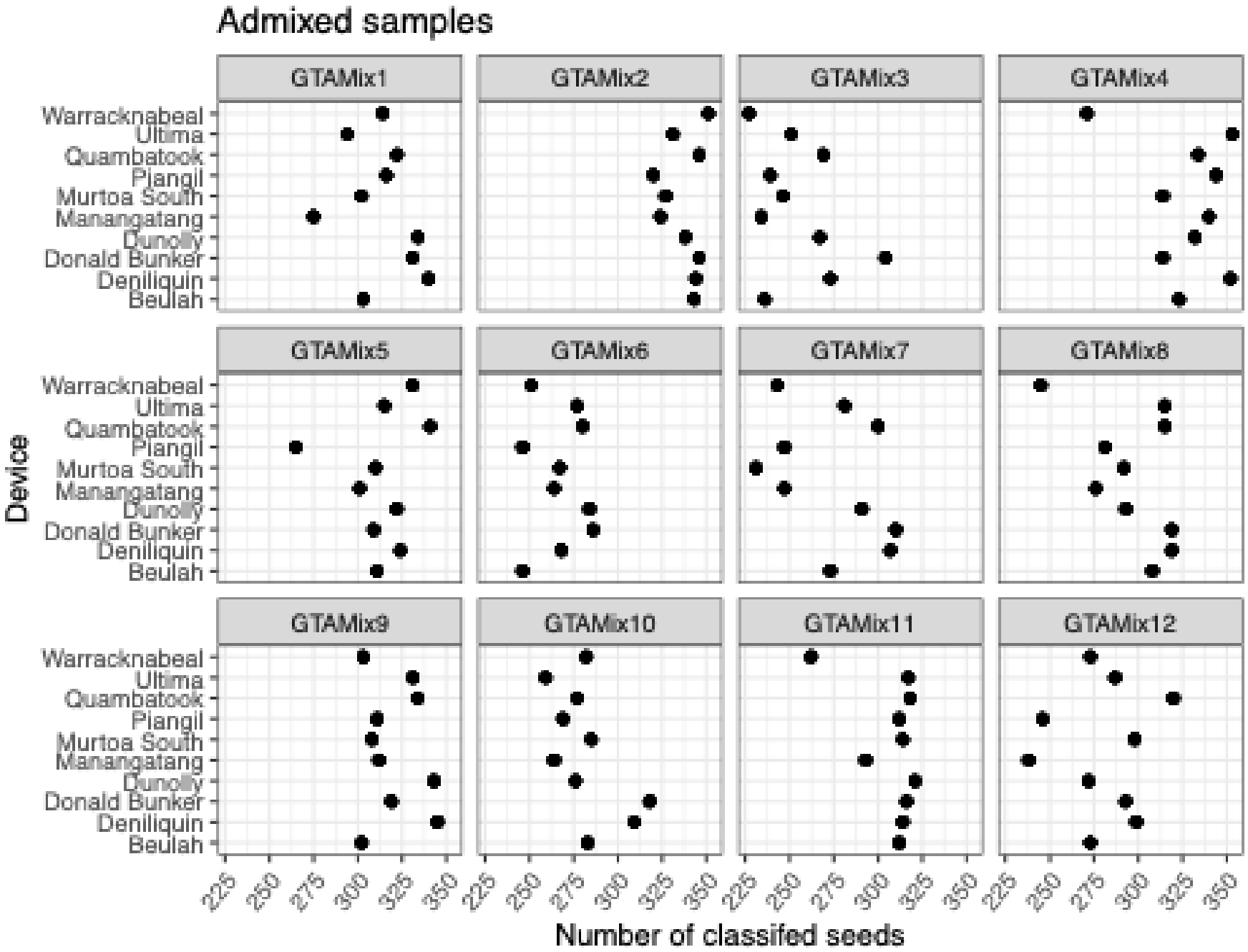
Consistency 🎯 **:** Show agreement across all devices and samples

Data viz 🎨

Allows for comparisons:

- within device across sample
- within sample

→ need to account for **variation in classified seeds** in analysis



summary

- Take your time to sniff + describe your data
- Keep question in mind
- Create your tidy data for downstream analyses
- Explore visually with your question in mind
- Breakdown your question