SUPPLEMENTAL APPENDIX:

Modifier to logMAR Verana Logic:

```
            if (snellen_denom is not None) & (mod_value is not None):
                if (snellen_denom in ["100", "114", "125", "150", "160", "200"]) & (mod_value
!= "1"):
                    if mod_op == "+":
                        if snellen_denom == "100":
                            new_denom = "80"
                        elif snellen_denom in ["114", "125"]:
                            new_denom = "100"
                        elif snellen_denom in ["150", "160"]:
                            new_denom = "125"
                        elif snellen_denom == "200":
                            new_denom = "160"
                    elif mod_op == "-":
                        if snellen_denom == "100":
                            new_denom = "125"
                        elif snellen_denom in ["114", "125"]:
                            new_denom = "160"
                        elif snellen_denom in ["150", "160", "200"]:
                            new_denom = "200"
                    else:
                        new_denom = snellen_denom
                    snellen = snellen_num + "/" + new_denom
                elif (snellen_denom in ["70", "63", "60", "50", "40", "32", "30", "25", "20", "16",
"12.5", "10"]) & (
                        mod_value not in ["1", "2"]):
                    if mod_op == "+":
                        if snellen_denom in ["70", "63", "60"]:
                            new_denom = "50"
                        elif snellen_denom == "50":
                            new_denom = "40"
                        elif snellen_denom == "40":
                            new_denom = "32"
                        elif snellen_denom in ["32", "30"]:
                            new_denom = "25"
                        elif snellen_denom == "25":
                            new_denom = "20"
                        elif snellen_denom == "20":
                            new_denom = "16"
                        elif snellen_denom == "16":
                            new_denom = "12.5"
                        elif snellen_denom in ["12.5", "10"]:
                            new_denom = "10"
                    elif mod_op == "-":
                        if snellen_denom in ["70", "63", "60"]:
                            new_denom = "80"
                        elif snellen_denom == "50":
                            new_denom = "63"
                        elif snellen_denom == "40":
                            new_denom = "50"
                        elif snellen_denom == "32":
                            new_denom = "40"
                        elif snellen_denom == "30":
                            new_denom = "30"
                        elif snellen_denom == "25":
                            new_denom = "32"
                        elif snellen_denom == "20":
                            new_denom = "25"
                        elif snellen_denom == "16":
                            new_denom = "20"
                        elif snellen_denom == "12.5":
                            new_denom = "16"
                        elif snellen_denom == "10":
                            new_denom = "12.5"
                        else:
                            new_denom = snellen_denom
                    snellen = snellen_num + "/" + new_denom
                elif (snellen_denom == "80"):
                    if (mod_value == "2") & (mod_op == "-"):
                        snellen = snellen_num + "/" + "100"
                    elif (mod_value in ["3", "4", "5", "6", "7", "8", "9"]) & (mod_op == "+"):
                        snellen = snellen_num + "/" + "63"
                    elif (mod_value in ["3", "4", "5", "6", "7", "8", "9"]) & (mod_op == "-"):
                        snellen = snellen_num + "/" + "100"
                    else:
                        snellen = distance_cleaned
                else:
                    snellen = distance_cleaned
```

LogMar lookup dictionary:

```
va_value_lookup = {
    "20/10":"-0.3", "20/12.5":"-0.2", "20/15":"-0.12", "20/16":"-0.1",
    "20/20":"0.0", "20/25":"0.1", "20/30":"0.18", "20/32":"0.2",
    "20/40":"0.3", "20/50":"0.4", "20/60":"0.48", "20/63":"0.5",
    "20/65":"0.51", "20/70":"0.54", "20/80":"0.6", "20/100":"0.7",
    "20/120":"0.78", "20/125":"0.8", "20/150":"0.88", "20/160":"0.9",
    "20/200":"1.0", "20/250":"1.1", "20/300":"1.18", "20/320":"1.2",
    "20/350":"1.24", "20/400":"1.3", "20/500":"1.4", "20/600":"1.48",
    "20/630":"1.5", "20/650":"1.51", "20/800":"1.6", "20/1000":"1.7",
    "20/1200":"1.78", "20/1260":"1.8", "20/1600":"1.9", "20/2000":"2.0",
    "J1":"0.0", "J2":"0.1", "J3":"0.18", "J4":"0.2", "J5":"0.3",
    "J6":"0.4", "J7":"0.48", "J8":"0.50", "J9":"0.6", "J10":"0.7",
    "J11":"0.76", "J12":"0.80", "J13":"0.90", "J14":"1.00", "NLP":"4",
    "HM":"2.3", "CF":"1.9", "LP":"2.7", "NI":"999"}
```

Rome Visual Acuity PostgreSQL filter:

```
SELECT
  patient_guid,
  eye,
  MIN(CAST(logmar AS DOUBLE PRECISION)) AS logmar,
  result_date,
  CAST(SUBSTRING(result_date, 1, 4) AS INTEGER) as result_year
FROM
  rome.patient_result_va
WHERE
  CAST(SUBSTRING(result_date, 1, 4) AS INTEGER) = 2018
  AND logmar > -1
  AND logmar <= 4
  AND (va_type = 1 or va_type = 999 or refraction = 'true')
  AND (va_method = 1 OR va_method = 999)
GROUP BY
  patient_guid,
  eye,
  result_date;
```

Chicago Visual Acuity PostgreSQL filter:

```
WITH
  logmar_t AS (
```

```sql
SELECT
  vh_patient_observation_uid,
  patient_guid,
  laterality_code,
  observation_date
FROM
  chicago_amc_2021_12_24.patient_observation
WHERE
  cluster_code = 'VA'
GROUP BY
  vh_patient_observation_uid,
  patient_guid,
  laterality_code,
  observation_date),
logmar_with_meta AS (
SELECT
  vh_patient_observation_uid,
  patient_guid,
  laterality_code,
  observation_date,
  MAX(distance) AS distance,
  MAX(correction) AS correction,
  MAX(refraction) AS refraction,
  MAX(pinhole) AS pinhole,
  MAX(logmar) AS logmar
FROM (
  SELECT
    logmar_t.vh_patient_observation_uid,
    patient_guid,
    laterality_code,
    observation_date,
    CASE
      WHEN modifier_group = 'chart_distance' AND modifier_value = 'distance' THEN 1
      WHEN modifier_group = 'chart_distance'
    AND modifier_value = 'near' THEN 2
      WHEN modifier_group = 'chart_distance' AND modifier_value = 'unknown' THEN 3
    ELSE
    0
  END
    AS distance,
    CASE
      WHEN modifier_group = 'correction' AND modifier_value = 'corrected' THEN 1
      WHEN modifier_group = 'correction'
    AND modifier_value = 'uncorrected' THEN 2
      WHEN modifier_group = 'correction' AND modifier_value = 'unknown' THEN 3
    ELSE
    0
  END
```

```
   AS correction,
   CASE
    WHEN modifier_group = 'refraction' AND modifier_value = 'true' THEN 1
    WHEN modifier_group ='‘refraction'
   AND modifier_value = 'false' THEN 2
   ELSE
   0
  END
   AS refraction,
   CASE
    WHEN modifier_group = 'pinhole' AND modifier_value = 'true' THEN 1
    WHEN modifier_group = 'pinhole'
   AND modifier_value = 'false' THEN 2
   ELSE
   0
  END
   AS pinhole,
   CASE
    WHEN modifier_group = 'logmar' THEN ROUND(CAST(modifier_value AS FLOAT), 2)
   ELSE
   -1.0
  END
   AS logmar
  FROM
   logmar_t
  JOIN
   chicago_amc_2021_12_24.patient_observation_modifier
  ON
   logmar_t.vh_patient_observation_uid = patient_observation_modifier.vh_patient_observation_uid)
 GROUP BY
  vh_patient_observation_uid,
  patient_guid,
  laterality_code,
  observation_date,
  logmar )
SELECT
 *
 FROM
  logmar_with_meta;
```