**Datasets loaded into PostreSQL:**

1. SSURGO data – Polygon shapefile
2. California National Highway System Dataset – Polyline shapefile – Downloaded from [California State Geoportal](https://gis.data.ca.gov/datasets/8887c21dce1d45ad93cd78a10b24781b_0?geometry=-146.159%2C31.074%2C-92.546%2C43.279)
3. Wildfire data – Polygon shapefile - obtained from KWG project
4. Distribution Centers – simulated point shapefile

**Query Set**

1. Natural Language – (Part-1) Where are [corn-suitable]-agricultural areas within ***20*** miles of ***Whittier*** in California? (Part-2) Are these areas currently being affected by wildfire?

Usefulness - Determine the impact in the price and availability of corn in the region  
surrounding ***Whittier.***

SPARQL Query 1 – SELECT ?soilarea

WHERE {  
:boundary.1324 geo:defaultGeometry ?aGeom .   
?aGeom geo:asWKT ?aWKT .  
?soilarea a :Soil\_Area.  
?soilarea :cornSuitabilityRating ?????.  
?soilarea geo:defaultGeometry ?bGeom .  
?bGeom geo:asWKT ?bWKT .

FILTER(geo:intersects(geo:buffer(?awkt,20,uom:miles),?bwkt))}

Query Complexity – 1 qualitative (poly-poly intersection) + 1 quantitative (buffer) spatial filters

SPARQL Query 2 – SELECT ?soilarea

WHERE {  
:boundary.1324 geo:defaultGeometry ?aGeom .   
?aGeom geo:asWKT ?aWKT .  
?soilarea a :Soil\_Area.  
?soilarea :cornSuitabilityRating ?????.  
?soilarea geo:defaultGeometry ?bGeom .  
?bGeom geo:asWKT ?bWKT .

?y a :Wildfire.

?y :eventDate ?fireDate.

?y geo:defaultGeometry ?cGeom .  
?cGeom geo:asWKT ?cWKT .

FILTER(geo:intersects(geo:buffer(?awkt,20,uom:miles),?bwkt))

FILTER(?fireDate > ‘?????’)

FILTER(geo:intersects(?bwkt, ?cwkt))}

Query Complexity – 2 qualitative (poly-poly intersection) + 1 quantitative (buffer) spatial filters + 1 regular filter

1. Natural Language – Where are the farmlands/agricultural regions within ***"Napa County / Solano County"*** county that had more than 5 wildfire incidents in the last ***3*** years?

SPARQL Query 3 – SELECT ?soilarea

WHERE {  
:boundary.166 geo:defaultGeometry ?aGeom .   
?aGeom geo:asWKT ?aWKT .  
?soilarea a :Soil\_Area.  
?soilarea geo:defaultGeometry ?bGeom .  
?bGeom geo:asWKT ?bWKT .

FILTER(geo:contains(?awkt, ?bwkt)

(SELECT (count(distinct ?w) as ?cw)

WHERE {

?w a :Wildfire.

?w :eventDate ?wDate.

?w geo:defaultGeometry ?wGeom .   
?wGeom geo:asWKT ?wWKT .

FILTER(?wDate > ???? AND ?wDate < ????)

FILTER(geof:intersects(?aWKT, ?wWKT))})

FILTER(?cw > 5)}

Query Complexity – Nested subquery with 2 filters (1 qualitative – poly-poly intersection), outer query has 1 qualitative (poly-poly intersection) + 1 regular filter

1. Natural Language – Where are the farmlands/agricultural regions within ***Napa*** ***county*** that has high crop productivity index, and have they had any wildfire incidents in the last ***3*** years?

Usefulness - Determine where the good farmland areas are that get affected by wildfire – maybe take extra precautionary measure to be proactive and prevent/control crop damage***.***

SPARQL Query 4 – SELECT ?soilarea

WHERE {  
:boundary.166 geo:defaultGeometry ?aGeom .   
?aGeom geo:asWKT ?aWKT .  
?soilarea a :Soil\_Area.  
?soilarea geo:defaultGeometry ?bGeom .  
?bGeom geo:asWKT ?bWKT .

?soilarea :cropProductivityIndex ?i.

FILTER(?i > ?????)

FILTER(geo:contains(?awkt, ?bwkt))}

Query Complexity – 1 qualitative (poly-poly intersection) + 1 regular filter

SPARQL Query 5 – SELECT ?soilarea

WHERE {  
:boundary.166 geo:defaultGeometry ?aGeom .   
?aGeom geo:asWKT ?aWKT .  
?soilarea a :Soil\_Area.  
?soilarea geo:defaultGeometry ?bGeom .  
?bGeom geo:asWKT ?bWKT .

(SELECT ?soilarea

WHERE {

?w a :Wildfire.

?w :eventDate ?wDate.

?w geo:defaultGeometry ?wGeom .   
?wGeom geo:asWKT ?wWKT .

?soilarea a :Soil\_Area.  
?soilarea geo:defaultGeometry ?bGeom .  
?bGeom geo:asWKT ?bWKT .

FILTER(?wDate > ???? AND ?wDate < ????)

FILTER(geof:intersects(?bWKT, ?wWKT))})

FILTER(geo:contains(?awkt, ?bwkt)}

Query Complexity – Nested subquery with 2 filters (1 qualitative – poly-poly intersection), outer query has 1 qualitative (poly-poly intersection) + 1 regular filter

1. Natural Language – Count the number of food distribution centers within ***Napa County***?

SPARQL Query 6 – SELECT (count(distinct ?d) as ?cd)

WHERE {

?d a :PointOfInterest.

?d :pointOfInterestCategory ‘Distribution Center’.

?d geo:defaultGeometry ?dGeom .   
?dGeom geo:asWKT ?dWKT .

?x a :Administrative\_Boundary.

?x geo:defaultGeometry ?xGeom .   
?xGeom geo:asWKT ?xWKT .

FILTER(geof:contains(xWKT, ?dWKT))}

Query Complexity – 1 qualitative (point in polygon) filter

1. Natural Language – Find the closest corn producing areas near ***distribution center 57056*** and find the distance between this distribution center and the closest road network?

SPARQL Query 7 – SELECT ?pName (MIN(?d) AS ?minValue)

WHERE {

:poi.57056 :pointOfInterestName ?pName

:poi.57056 geo:defaultGeometry ?aGeom .

?aGeom geo:asWKT ?aWKT .

?y a :Soil\_Area.

?y :cornSuitabilityArea ????.

?y geo:defaultGeometry ?bGeom .

?bGeom geo:asWKT ?bWKT .

BIND(geof:distance(?aWKT, ?bWKT,uom:metre) as ?d)}

GROUP BY ?pName

ORDER BY ASC(?minValue)

LIMIT 3

Query Complexity – 1 quantitative (distance between point and polygon) filter + GROUP BY, Order, LIMIT

SPARQL Query 8 – SELECT ?rName (MIN(?d) AS ?minValue)

WHERE {

:poi.57056 :pointOfInterestName ?pName

:poi.57056 geo:defaultGeometry ?aGeom .

?aGeom geo:asWKT ?aWKT .

?y a :Road\_Element.

?y geo:defaultGeometry ?bGeom .

?bGeom geo:asWKT ?bWKT .

BIND(geof:distance(?aWKT, ?bWKT,uom:metre) as ?d)}

GROUP BY ?rName

ORDER BY ASC(?minValue)

LIMIT 3

Query Complexity – 1 quantitative (distance between point and polygon) filter + GROUP BY, Order, LIMIT