**ni\_nearest\_point\_to\_line\_segment** – this function splits all line features by a given distance (shorter segments are not extended, whilst longer segments are split), then creates a buffer around each resultant line segment and finally finds the closest point within each buffer of each line segment. The resultant output table will contain all the attributes of both the original line data for each segment, as well as the point attributes of the closest point within each buffer to each line segment, as well as the following:

* Point\_prkey – point table primary key
* Point\_geom – input point geometry
* Point\_fraction – fraction of a line along which a point lies
* Perp\_point\_on\_line – point on the line
* Point\_to\_perp\_point\_line – new geometry linking line segment with nearest point to line segment
* Point\_to\_perp\_point\_line\_length – length of *“point\_to\_perp\_point\_line”*

**Parameters:**

* Line\_table\_name: string - Table name of line dataset
* Line\_table\_geometry\_column\_name: string - Geometry column name of line dataset
* Line\_table\_prkey: string - Line dataset primary key column
* Line\_geom\_prefix: string - Prefix to apply to all attributes of the line dataset (used to avoid duplicate column names in final output)
* Point\_table\_name: string - Table name of point dataset
* Point\_table\_geometry\_column\_name: string - Geometry column name of point dataset
* Point\_table\_prkey: string - Point dataset primary key column
* Point\_geom\_prefix: string - Prefix to apply to all attributes of the point dataset (used to avoid duplicate column names in final output)
* Line\_split\_distance: float - Distance by which to split the line dataset
* Buffer\_distance: float - Buffer distance for each line segment
* Output\_table\_name: string - Output table name
* Add\_to\_geometry\_columns: Boolean - Boolean to denote whether or not to add the geometry column name of the resultant output table to the geometry column table (if true, adds point\_geom for point geometry, line\_geom for line geometry, and point\_to\_perp\_point\_line for a line that joins the closest point in each segment to each line)

For example the following function could be used to find the closest electricity sub station per section of rail:

e.g. SELECT \* FROM ni\_nearest\_point\_to\_line\_segment('os\_meridian\_2\_rail\_ln\_polyline', 'geom', 'code', 'linedata\_', 'OS\_ElectricitySubStations', 'geom', 'gid', 'pointdata\_', 2000, 500, 'closest\_os\_electricitysubstation\_to\_electrified\_rail\_sections', false) f(point\_prkey integer, point\_geom geometry, point\_fraction double precision, perp\_point\_on\_line geometry, point\_to\_perp\_point\_line geometry, point\_to\_perp\_point\_line\_length double precision, line\_prkey integer, line\_geom geometry, linedata\_code integer, linedata\_identifier char(13), linedata\_name char(70), linedata\_geom geometry, pointdata\_gid integer, pointdata\_oid\_ varchar(254), pointdata\_os\_toid varchar(254), pointdata\_easting numeric, pointdata\_northing numeric, pointdata\_geom geometry);

|  |
| --- |
|  |
| Schematic of underlying process of the **ni\_nearest\_point\_to\_line\_segment** function. This could be used to calculate the likely substations supplying power to the electrified sections of the rail network, in the absence of real data. |

|  |  |
| --- | --- |
|  |  |
| Finding substations along sections of National Grid Gas Transmission Network | National Grid Gas Transmission Network (green line), electricity substations (yellow dot), link between substation and sections of gas network (red dashed line) |