The errors in the Terraform execution are due to `null` or empty values being passed to functions that expect non-null arguments. Here's how to troubleshoot and fix them:

### \*\*1st Error: `coalescelist` Failed Due to `null` Arguments\*\*

```

Error: Error in function call

on .terraform/modules/eks/main.tf line 48, in resource "aws\_eks\_cluster" "this":

48: subnet\_ids = coalescelist(var.control\_plane\_subnet\_ids, var.subnet\_ids)

while calling coalescelist(vals...)

var.control\_plane\_subnet\_ids is null

var.subnet\_ids is null

Call to function "coalescelist" failed: no non-null arguments.

```

#### \*\*Cause:\*\*

The `coalescelist` function is failing because both `var.control\_plane\_subnet\_ids` and `var.subnet\_ids` are `null`. `coalescelist` requires at least one non-null argument.

#### \*\*Solution:\*\*

- Ensure that these variables are properly defined in your `terraform.tfvars` or module input variables.

- Provide default values for these variables if they might be `null`, for example:

```hcl

variable "control\_plane\_subnet\_ids" {

type = list(string)

default = []

}

variable "subnet\_ids" {

type = list(string)

default = []

}

```

- If these values should be coming from a data source, ensure the data source is returning expected values before referencing them.

---

### \*\*2nd Error: `element` Function Failed Due to an Empty List\*\*

```

Error: Error in function call

on .terraform/modules/vpc/main.tf line 1098, in resource "aws\_eip" "nat":

1098: element(var.azs, var.single\_nat\_gateway ? 0 : count.index)

while calling element(list, index)

count.index is 0

var.azs is empty list of string

var.single\_nat\_gateway is true

Call to function "element" failed: cannot use element function with an empty list.

```

#### \*\*Cause:\*\*

The `element` function is trying to access an index in `var.azs`, but `var.azs` is an empty list. Terraform does not allow fetching an element from an empty list.

#### \*\*Solution:\*\*

- Ensure `var.azs` is populated correctly in your configuration.

- Provide a default value for `var.azs` if necessary:

```hcl

variable "azs" {

type = list(string)

default = ["us-east-1a", "us-east-1b", "us-east-1c"] # Adjust based on your region

}

```

- If `var.azs` should be coming from a data source (e.g., `data.aws\_availability\_zones.available.names`), confirm that it returns valid values.

---

### \*\*Next Steps\*\*

1. \*\*Check Variable Definitions\*\*

Run:

```bash

terraform console

```

Then type:

```hcl

var.azs

var.control\_plane\_subnet\_ids

var.subnet\_ids

```

This will confirm if they are empty or undefined.

2. \*\*Check Terraform State or Data Sources\*\*

If these values are supposed to be populated from AWS data sources, verify their outputs with:

```hcl

data "aws\_subnet\_ids" "selected" {

vpc\_id = var.vpc\_id

}

output "selected\_subnet\_ids" {

value = data.aws\_subnet\_ids.selected.ids

}

```

Run `terraform apply` and check the output.

3. \*\*Ensure Required Variables are Set\*\*

If using `terraform.tfvars`, check that it contains the necessary values:

```hcl

control\_plane\_subnet\_ids = ["subnet-1234", "subnet-5678"]

subnet\_ids = ["subnet-1234", "subnet-5678"]

azs = ["us-east-1a", "us-east-1b"]

```

If you're still having issues, let me know how these variables are being set in your configuration.